

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

FREE ASSOCIATIVE NEURONS – FAN
UMA ABORDAGEM PARA RECONHECIMENTO DE
PADRÕES

DISSERTAÇÃO SUBMETIDA À UNIVERSIDADE FEDERAL DE SANTA CATARINA
PARA OBTENÇÃO DO TÍTULO DE MESTRE EM ENGENHARIA DE PRODUÇÃO.



ROBERTO TADEU RAITTZ



UFSC-BU

ORIENTADOR: PROF. FERNANDO ÁLVARO OSTUNI GAUTHIER, DR.

FLORIANÓPOLIS, DEZEMBRO DE 1997

FREE ASSOCIATIVE NEURONS – FAN
UMA ABORDAGEM PARA RECONHECIMENTO DE
PADRÕES

ROBERTO TADEU RAITTZ

ESTA DISSERTAÇÃO FOI JULGADA ADEQUADA PARA A OBTENÇÃO DO TÍTULO DE MESTRE EM ENGENHARIA DE PRODUÇÃO E A PROVADA EM SUA FORMA FINAL PELO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO.



PROF. RICARDO MIRANDA BARCIA, PH.D.
COORDENADOR DO CURSO

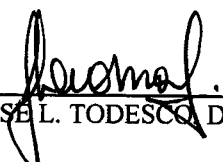
BANCA EXAMINADORA:




PROF. FERNANDO Á. O. GAUTHIER, Dr.
Orientador



PROF. ALEJANDRO M. RODRIGUEZ, Dr.



PROF. JOSÉ L. TODESCO, Dr.



PROF. ROBERTO C. S. PACHECO, Dr.

*Aos três iniciados que me ensinaram a
recuperar o meu amor pela ciência.*

À Mônica.

AGRADECIMENTOS

Não é possível agradecer a tantos quantos deveria, que de forma direta ou indireta são fundamentais para a existência deste trabalho.

No entanto não posso deixar de agradecer especialmente a algumas pessoas que participaram de forma muito próxima, de todas as etapas do desenvolvimento do tema desta dissertação. Essas pessoas são:

O Orientador

Prof. Dr. Fernando Álvaro Ostuni Gauthier

Os Professores

Prof. Dr. José Leomar Todesco

Prof. Dr. Alejandro Martins

Prof. Dr. Roberto Carlos Pacheco

Os colegas

Paulo Afonso Bracarense Costa

Gertrudes Aparecida Dandolini

João Arthur de Souza

Mauro José Belli

Agradeço também às instituições

A UFPR e especialmente a ET-UFPR

A CAPES pelo apoio financeiro

E a todas as pessoas da minha família, por tudo...

RESUMO

Nesta dissertação apresenta-se um novo método de aprendizagem: Free Associative Neurons (FAN). FAN é constituído por neurônios independentes com capacidade autônoma de aprendizado. O poder de aprendizado de FAN está baseado na granularidade na representação da informação. O método é embasado na lógica difusa e nas noções de redes neuronais. Sendo um método para representação de ambientes complexos, FAN pode ser aplicado no reconhecimento de padrões, classificação e diagnóstico. A aplicação que este trabalho trata é o Reconhecimento de Padrões. Apresentam-se os princípios que regem FAN e também são discutidos os resultados obtidos pelo método e sua comparação com outros algoritmos conhecidos de redes neuronais. Na conclusão são apresentadas propostas para trabalhos futuros.

ABSTRACT

In this thesis is introduced a new learning method: Free Associative Neurons (FAN). FAN is composed by independent neurons with autonomous learning capability. The learning power of FAN is based on the use of granularity for representing information. Its basis is the Fuzzy logic and it is also loosely based on the notion of the Neural Networks. As a method for representing complex environments, FAN can be used in Pattern Recognition, Classification and Diagnosis. The approach that is the subject of this work is the use of FAN for Pattern Recognition. It is focused on the principles governing FAN and also discusses some results obtained and its comparison with well-known Artificial Neural Network algorithms. In the conclusion some future works are proposed.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Considerações iniciais	1
1.2	Objetivos	3
1.3	Justificativa	4
1.4	Limitações	4
1.5	Estrutura	5
2	CONJUNTOS DIFUSOS	7
2.1	Introdução	7
2.2	Visão geral da lógica difusa	7
2.3	Conjuntos difusos	8
2.4	Formação de conjuntos difusos (definição das funções de pertinência)	10
2.5	Representação de conjuntos difusos	11
2.6	Representação adicional de vaguesa pelo uso de advérbios (Hedges)	13
2.7	Operadores difusos	14
2.7.1	Intersecção	15
2.7.2	União	16
2.7.3	Complementação	16
2.8	Conjuntos difusos e FAN	17
2.9	Conclusão	18
3	RECONHECIMENTO DE PADRÕES	19
3.1	Introdução	19

3.2	Definição de Reconhecimento de Padrões (RP)	19
3.3	Estrutura típica de um sistema de reconhecimento de padrões	21
3.4	Aplicações do Reconhecimento de Padrões (RP)	21
3.5	Abordagens em RP	23
3.5.1	Introdução	23
3.5.2	Reconhecimento de padrões estatístico	23
3.5.3	Reconhecimento de padrões sintático	24
3.5.4	Reconhecimento de padrões por redes neurais	24
3.5.5	Reconhecimento de padrões usando conjuntos difusos	26
4	MODELO PROPOSTO	28
4.1	Introdução	28
4.2	Problema	29
4.3	Modelo proposto (FAN)	30
4.3.1	Modelagem dos dados	30
4.4	Método de Aprendizado FAN	34
4.5	Algoritmo FAN	38
4.6	Conclusão	40
5	APLICAÇÕES	42
5.1	Introdução	42
5.2	Primeira fase - ÍRIS DE FISHER	44
5.2.1	Definição do problema	44
5.2.2	Comentários	44
5.2.3	Resultados e comentários	45
5.3	Segunda fase - Classificação de cromossomas no grupo de Denver	46
5.3.1	Definição do problema	46
5.3.2	Comentários	46
5.3.3	Resultados e comparações	47

5.4	Terceira fase - Saúde financeira de pequenas empresas	48
5.4.1	Definição do problema	48
5.4.2	Comentários	49
5.4.3	Resultados e comparações	50
5.5	Experimentos	51
5.5.1	Experimentos relacionados à confusão entre as classes	51
5.5.2	Experimentos relacionados ao número de épocas	60
6	CONCLUSÃO	62
	REFERÊNCIAS BIBLIOGRÁFICAS	65

LISTA DE FIGURAS

FIGURA 2.1	CONJUNTOS DIFUSO E CLÁSSICO DE CRIANÇAS	9
FIGURA 2.2	EXEMPLO DE FUNÇÃO DE PERTINÊNCIA TRIANGULAR	13
FIGURA 3.1	SISTEMA TÍPICO DE RECONHECIMENTO DE PADRÕES	21
FIGURA 4.1	VISÃO ESQUEMÁTICA DO PROCESSO DE MODELAGEM DE DADOS	35
FIGURA 4.2	PROJEÇÃO DA VIZINHANÇA DIFUSA	35
FIGURA 4.3	INTERCECÇÃO ENTRE F E FAN ^J *	38
FIGURA 5.1	GRÁFICO DE DESEMPENHO (TAXA DE ERRO X NÚMERO DE ÉPOCAS)	61

LISTA DE TABELAS

TABELA 4.1	EXEMPLO DA PROJEÇÃO DE UM VIZINHO	34
TABELA 5.1	COMPARATIVO ENTRE TAXAS DE ERROS (TESTE)	48
TABELA 5.2	COMPARATIVO ENTRE TAXAS DE ERROS (TREINAMENTO E TESTE)	51

LISTA DE QUADROS

QUADRO 5.1	TESTE REALIZADO COM TREINAMENTO COMPLETO SOBRE O PRÓPRIO CONJUNTO DE TREINAMENTO	53
QUADRO 5.2	TREINAMENTO COMPLETO APLICADO AO CONJUNTO DE TESTE	53
QUADRO 5.3	TESTE REALIZADO COM TREINAMENTO COM 2100 PADRÕES, 300 POR CLASSE, APLICADO SOBRE O PRÓPRIO CONJUNTO DE TREINAMENTO	54
QUADRO 5.4	TREINAMENTO COM 300 PADRÕES PARA CADA CLASSE, TESTADO NO CONJUNTO COMPLETO DE TESTE	54
QUADRO 5.5	TESTE REALIZADO COM 200 PADRÕES PARA CADA CLASSE, SOBRE O PRÓPRIO CONJUNTO DE TREINAMENTO	55
QUADRO 5.6	TESTE REALIZADO COM 200 PADRÕES PARA CADA CLASSE, SOBRE O CONJUNTO DE TESTE.	55
QUADRO 5.7	TESTE REALIZADO COM 100 PADRÕES PARA CADA CLASSE, SOBRE O PRÓPRIO CONJUNTO DE TREINAMENTO	56
QUADRO 5.8	TESTE REALIZADO COM 100 PADRÕES PARA CADA CLASSE, SOBRE O PRÓPRIO CONJUNTO DE TESTE	56
QUADRO 5.9	TESTE REALIZADO COM 200 PADRÕES NAS CINCO PRIMEIRAS CLASSES E 300 NAS DUAS ÚLTIMAS, SOBRE O PRÓPRIO CONJUNTO DE TREINAMENTO	57
QUADRO 5.10	TESTE REALIZADO COM 200 PADRÕES EM CADA UMAS DAS CINCO PRIMEIRAS CLASSES E 300 PARA AS DUAS ÚLTIMAS, SOBRE O CONJUNTO DE TESTE	57
QUADRO 5.11	TESTE REALIZADO COM A DISTRIBUIÇÃO DE PADRÕES PROPORCIONAL À DISTRIBUIÇÃO DO CONJUNTO DE TREINAMENTO COMPLETO, SOBRE O PRÓPRIO CONJUNTO DE TREINAMENTO PADRÕES PARA CADA CLASSE, SOBRE O PRÓPRIO CONJUNTO DE TESTE	58

QUADRO 5.12	TESTE REALIZADO COM A DISTRIBUIÇÃO DE PADRÕES PROPORCIONAL À DISTRIBUIÇÃO DO CONJUNTO DE TREINAMENTO COMPLETO, SOBRE O CONJUNTO DE TESTE	58
-------------	--	----

1 Introdução

1.1 Considerações iniciais

O Curso de Pós-Graduação em Engenharia de Produção da Universidade Federal de Santa Catarina, e particularmente a área de concentração em inteligência aplicada, possuem algumas características que podem ser consideradas essenciais para o desenvolvimento deste trabalho.

Os alunos que freqüentam os diversos cursos ofertados são provenientes de uma grande variedade de cursos de formação básica, e os programas dos cursos são adaptados a esta diversidade proporcionando aos candidatos à pós-graduação um ambiente ricamente interdisciplinar, quer nas turmas onde realiza os créditos, quer nos grupos de trabalho formados para estudos e pesquisa freqüentemente formados por professores e alunos. Em adição a este aspecto, os conteúdos técnicos e científicos da inteligência artificial jamais são perdidos de vista em relação à rigorosidade com qual são tratados e à sua atualidade.

Foi nesse ambiente, que o conceito de Neurônio Livre Associativo (FAN-Free Associative Neuron) foi concebido e desenvolvido, por um grupo de alunos e professores do Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de Santa Catarina. Neste trabalho, o primeiro de uma série, são estabelecidos os conceitos fundamentais desta técnica, e é apresentada a sua adaptação para aplicações em reconhecimento de padrões.

No Reconhecimento de Padrões, a unidade FAN de reconhecimento é associada a uma classe específica. Todas as relações entre as características de uma determinada classe são estabelecidas unicamente na unidade que a representa, livre portanto das relações diretas com as demais classes do domínio do problema.

Uma unidade FAN é um esquema de combinações de características, associado a uma classe, e um conjunto de parâmetros resultantes da utilização desse esquema no seu treinamento.

Ao ser apresentado um novo padrão de treinamento, o FAN (unidade) pode ser treinado para reconhecê-lo/esquecê-lo, através dos reforços/enfraquecimentos nos seus parâmetros. No teste de um padrão, o FAN permite atribuir seu grau de pertinência em relação à classe que representa.

O processo de aprendizagem de uma rede FAN, para o reconhecimento de várias classes, é realizado treinando-se cada unidade com os padrões relacionados à classe que representa.

A discriminação da classe à qual um determinado padrão deve estar relacionado em um teste é feita através da escolha da classe cuja unidade FAN correspondente indicar maior grau de pertinência (similaridade sob um aspecto) para o referido padrão.

O argumento intuitivo para a escolha de uma abordagem como a proposta, é modelar a visão de conjunto. Objetos muito simples tendem a ser indistinguíveis e objetos com relações complicadas entre suas características podem ser identificados pelos seus detalhes únicos. Como um exemplo disto, tome-se como referências para identificação de um indivíduo primeiro a forma de

seu polegar e em seguida sua impressão digital. A análise deste exemplo, deve no que tange à capacidade de discriminação, dispensar comentários.

A visão que acompanha o desenvolvimento das unidades FAN é correspondente à ilustração anterior. Deve-se desenvolver uma estrutura matemática semelhante ao esquema de uma impressão digital, de grande complexidade interna, porém autocontida, que possa reunir as relações de características que as tornem uma fonte de relacionamento ótimo (único?) com a classe que representa.

A forma encontrada para este modelo e os resultados obtidos com a sua aplicação no reconhecimento de padrões é o objeto desta dissertação.

1.2 Objetivos

O objetivo geral deste trabalho, é apresentar um novo método para o reconhecimento de padrões, introduzindo o conceito de Neurônio Livre Associativo (em inglês Free Associative Neuron - FAN). Apresentam-se como objetivos específicos:

- trazer à tona a discussão sobre a necessidade de métodos genéricos de RP;
- propor uma abordagem singular para a extração de graus de pertinência em conjuntos difusos;
- discutir as vantagens em se utilizar unidades independentes para representar as classes em RP;
- mostrar como a expansão do espaço das características de um padrão, baseada na teoria dos conjuntos difusos, pode ser usada para se obter bons resultados em RP;
- abrir novos caminhos para a popularização das técnicas de RP.

1.3 Justificativa

O uso de técnicas para reconhecimento de padrões com o avanço da inteligência artificial, também evoluiu nos últimos tempos, porém ainda não atendem completamente à sua demanda. Ótimas soluções para problemas de classificação são obtidas facilmente nos meios acadêmicos, técnicos ou científicos. Por outro lado não encontramos aplicativos mais populares que possam ser utilizados por profissionais de áreas menos especializadas. Ao realizarem-se estudos de novas abordagens abrem-se mais caminhos para o surgimento de tais aplicativos, principalmente quando esta situação é posta como objeto de estudos.

Por outro lado, treinamento de unidades de representação de classes independentes é uma necessidade quando se tem em vista a generalidade que uma técnica de RP se propõe a apresentar, além de ser fundamental no relacionamento entre RP e bancos de dados, que é o caminho para aplicações mais flexíveis em processamento de dados.

1.4 Limitações

Esta dissertação não visa concluir os estudos que vêm sendo desenvolvidos a cerca de FAN, mostra uma abordagem, fiel à idéia inicial de sua concepção, que pretende introduzir na comunidade acadêmica a essência de FAN. Assim sendo, não se propõem a fazer uma revisão bibliográfica completa de todos os outros estudos relacionados, o que implicaria em um trabalho muito mais extenso.

Da mesma forma o desempenho computacional do método não é aprofundado. Ainda que tenham sido desenvolvidas e testadas várias implementações de FAN, elas não serão discutidas em detalhes como tempo de processamento utilização de memória, ou outros aspectos computacionais.

Não se trata de um estudo de caso de uma aplicação em um problema específico de RP, mas a apresentação de uma técnica genérica. Os problemas onde FAN foi utilizado e testado não são explicados, porém apresentamos a bibliografia onde podem ser encontrados.

1.5 Estrutura

O primeiro capítulo traz uma visão geral do trabalho, seus objetivos, justificativa, limitações e a estrutura.

No segundo capítulo é realizada uma revisão dos principais aspectos da teoria dos conjuntos difusos, e a explicação dos tópicos desta teoria usados no FAN.

O terceiro capítulo versa sobre a teoria de reconhecimento de padrões, com ênfase para o reconhecimento de padrões difuso.

O quarto capítulo traz o método proposto para reconhecimento de padrões usando FAN, seu desenvolvimento matemático e algorítmico.

O quinto capítulo é uma discussão das aplicações onde o método proposto foi testado e os resultados obtidos.

O sexto capítulo apresenta as conclusões do trabalho e propostas para trabalhos futuros. Finalmente é registrada a bibliografia utilizada nesse trabalho.

2 Conjuntos difusos

2.1 Introdução

Neste capítulo é feita uma revisão dos conceitos fundamentais da teoria dos conjuntos difusos relacionados à lógica difusa, seguindo a linha utilizada por [DURK1994], e acrescentados outros necessários para o desenvolvimento da unidade FAN extraídos de [ROSS1995] [KLIR1995] [KOSKO1992].

2.2 Visão geral da lógica difusa

O primeiro a utilizar conceitos de lógica difusa foi Lukasiewicz na década de 1920, ao estudar a representação matemática de termos como *alto*, *velho* ou *quente*. Tais termos não podem ser adequadamente representados pela lógica aristotélica, baseada na verdade ou falsidade absoluta, 0 ou 1. O matemático polonês desenvolveu um sistema de lógica, onde os graus de verdade possíveis se estendem aos números reais no intervalo $[0,1]$. Ele usou um número neste intervalo para representar a *possibilidade* de uma declaração dada ser verdadeira ou falsa. Por exemplo, a possibilidade de uma pessoa de 1,80 m de altura ser realmente alta pode ser relacionada ao valor 0,9; é muito provável que tal pessoa seja alta. Este raciocínio levou ao desenvolvimento de uma técnica formal para tratar o raciocínio inexato, denominado teoria da possibilidade.

Em 1965, Zadeh estendeu o trabalho a um sistema formal de lógica matemática. O trabalho de Zadeh possui uma relevância especial, por ter transformado em uma ferramenta que permite o uso prático nos meios técnicos e científicos de conceitos antes essencialmente filosóficos. Essa *ferramenta* valiosa para o tratamento de termos da linguagem natural é chamada Lógica Difusa.

Define-se a lógica difusa como o ramo da lógica que usa graus de pertinência em conjuntos, ao invés da pertinência/ausência absolutas: verdadeiro ou falso.

2.3 Conjuntos difusos

A teoria tradicional dos conjuntos, olha para o mundo como se este fosse preto e branco. Isto significa que um objeto pertence ou não pertence a um determinado conjunto. Se for considerado por exemplo o conjunto das crianças, a teoria clássica dos conjuntos usa a definição de um limite rígido de idade a partir da qual as pessoas deixam de ser crianças, sendo que todos os indivíduos com idade aquém deste limite são membros do conjunto com valor 1 e os primeiros o são com valor 0. Então se o limite de idade for estabelecido em 10 anos por exemplo, alguém ao completar onze anos deixa de ser a criança que fora no momento anterior.

A lógica difusa é mais razoável na interpretação de casos como o exposto acima. O uso de conjuntos difusos com graus de pertinência entre 0 e 1 para os seus membros a associação membro/conjunto é mais natural. Por exemplo, uma pessoa com 5 anos pode ser associada ao conjunto das crianças com o valor de pertinência 0,9; já uma de 13 anos com valor 0,1.

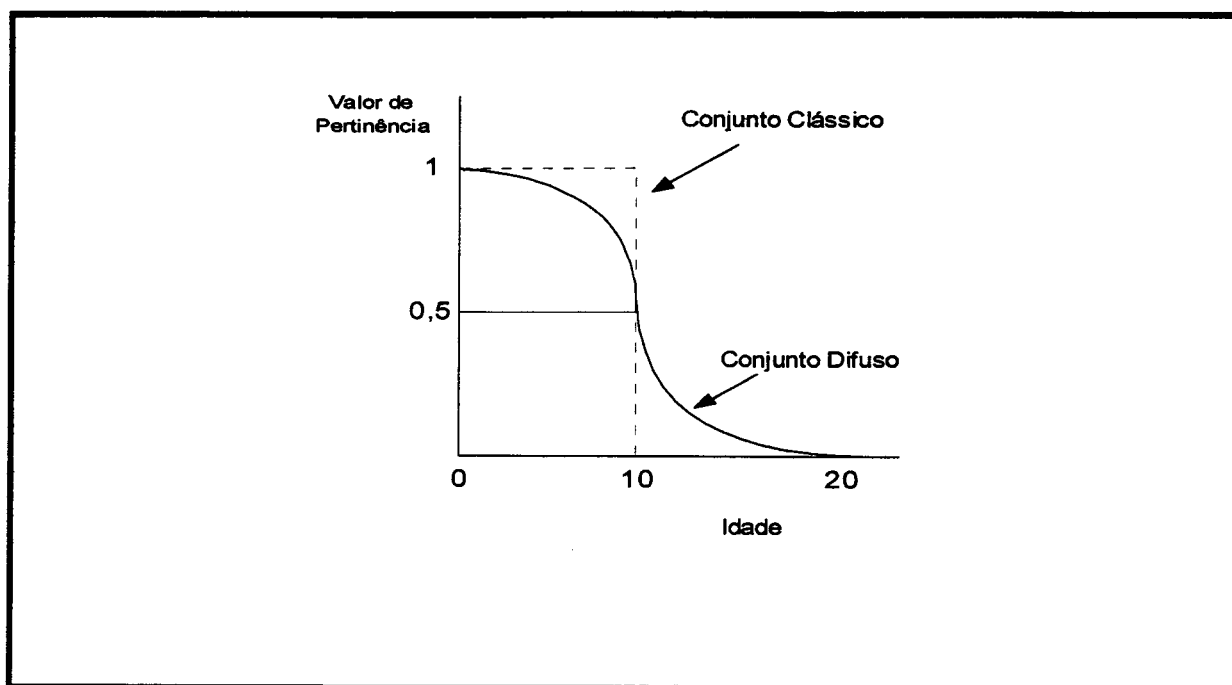


Figura 2.1 – Conjuntos difuso e clássico de crianças

Os conjuntos difusos proporcionam uma transição suave dos limites ilustrados na Figura 2.1. O eixo- x , definido como universo de discurso, representa as idades das pessoas. O eixo- y é o dos valores de pertinência ao conjunto difuso. Este conjunto, das crianças, mapeia as idades em relação aos seus valores de pertinência. Observando-se a figura, uma pessoa de 11 anos não deixa de ser criança subitamente. A pessoa vai deixando de pertencer ao conjunto das crianças lentamente conforme vai aumentando sua idade. A definição formal de conjunto difuso é a seguinte:

Seja X o universo de discurso, com os elementos de X denotados por x . Um conjunto difuso A de X é caracterizado por uma função de pertinência $\mu_A(x)$ que associa cada elemento x com um grau de pertinência em A .

Em contraste com a teoria da probabilidade, que atribui probabilidades a um dado evento baseadas na sua frequência de ocorrência, a

lógica difusa atribui valores aos eventos baseados na função de pertinência definida como:

$$\mu_A(x): X \rightarrow [0,1]$$

Na lógica difusa o evento ou elemento x é associado a uma função de pertinência μ . Esse valor representa o grau ao qual o elemento x pertence ao conjunto difuso A .

$$\mu_A(x) = \text{Grau}(x \in A)$$

O valor de pertinência de x é delimitado pela seguinte relação:

$$0 \leq \mu_A(x) \leq 1$$

Os conjuntos difusos são uma extensão da teoria tradicional dos conjuntos. Ela generaliza o conceito de pertinência, usando a função de pertinência μ que retorna um valor entre 0 e 1 que representa o grau de pertinência que um objeto x tem em relação ao conjunto A .

2.4 Formação de conjuntos difusos (definição das funções de pertinência)

Para representar-se conjuntos difusos em computadores é necessário definir suas funções de pertinência. Para fazê-lo é conveniente consultar pessoas que conhecem bem o domínio do problema a ser tratado, ou que cuja opinião sobre a significação da variável a ser modelada seja significativa para sua boa representação.

Numerosos métodos para construção de funções de pertinência se encontram descritos na literatura dos conjuntos difusos e quase que invariavelmente, são baseados em julgamentos de especialistas no assunto que se quer representar. Estes são classificados em duas categorias fundamentais: métodos diretos e métodos indiretos. Nos métodos diretos espera-se que o especialista responda questões de vários tipos que definam explicitamente as funções de pertinência. Nos métodos indiretos, as informações fornecidas pelos especialistas são mais genéricas e afetam as funções construídas apenas implicitamente [KLIR1995].

2.5 Representação de conjuntos difusos

A forma de representação dos conjuntos difusos utilizada pela literatura, possui algumas características que podem confundir o iniciante no estudo da teoria. Segue-se uma introdução sobre a representação formal dos conjuntos difusos.

Dado um universo de discurso X e um conjunto difuso A definido nele, e um conjunto de elementos discretos de X elementos $\{x_1, x_2, \dots, x_n\}$. O conjunto difuso A , define a função de pertinência $\mu_A(x)$, que mapeia os elementos x_i de X , em relação aos graus de pertinência em $[0,1]$. Os graus de pertinência indicam qual é o grau que x_i pertence a A . Para um conjunto discreto de elementos, uma representação = (conveniente é feita usando um vetor:

$$A = (a_1, a_2, \dots, a_n)$$

onde,

$$a_i = \mu_A(x_i)$$

Para uma representação mais clara, o vetor normalmente inclui o símbolo “/” que associa o valor de pertinência a_i com sua coordenada x_i :

$$A = (a_1/x_1, a_2/x_2, \dots, a_n/x_n)$$

É importante observar que o sinal “/” não representa fração, mas a relação entre valor e pertinência.

A notação padrão dos conjuntos difusos, representa a união das dimensões dos vetores como é mostrado na seqüência; onde “+” representa a notação de união Booleana:

$$A = \mu_1/x_1 + \mu_2/x_2 + \dots + \mu_n/x_n$$

ou

$$A = \sum_{i=1..n} \mu_i/x_i$$

Se X é uma função contínua, então o conjunto A pode ser representado da seguinte forma:

$$A = \int_x \mu_A(x_i) / x_i$$

Para um conjunto contínuo de elementos, é necessário usar funções para mapear os elementos em relação aos seus graus de pertinência. Algumas funções típicas são as sigmoidais, gaussianas, e pi, que são curvas suaves e que freqüentemente representam bem os dados. Porém sob o ponto de vista computacional nem sempre são as mais convenientes. Se a economia no tempo de processamento é mais relevante em uma aplicação que o critério da suavidade,

pode-se usar funções mais simples e em muitos casos eficientes como funções triangulares:

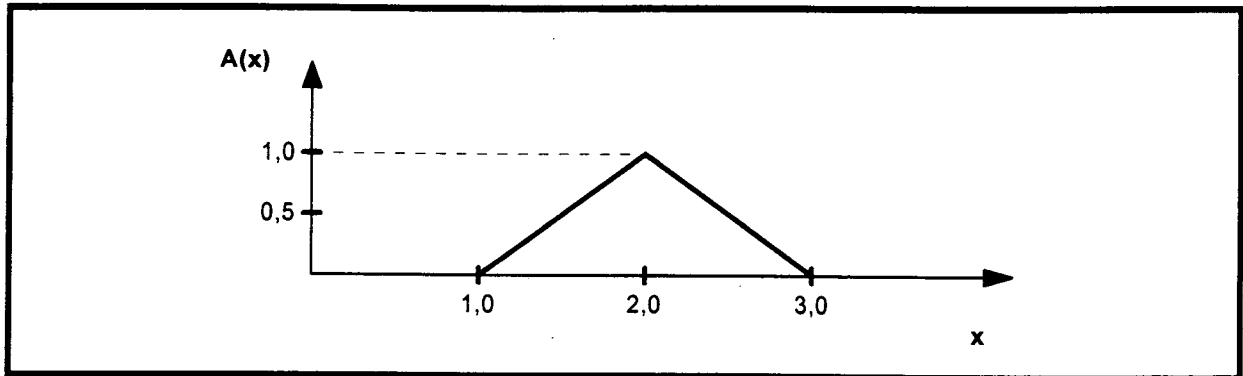


Figura 2.2 – Exemplo de função de pertinência triangular.

$$\begin{aligned}
 A(x) &= \begin{cases} p(x-r) + 1 & \text{quando } x \in [r - 1/p, r] \\ p(r-x) + 1 & \text{quando } x \in [r, r + 1/p] \\ 0 & \text{para outros casos} \end{cases}
 \end{aligned}$$

Onde r é o valor para o qual se deseja que o grau de pertinência seja 1 ($r=2$, no exemplo) e p é o parâmetro que determina a taxa com a qual, para cada x , a função cresce ou decresce, com o aumento da diferença $|r-x|$.

2.6 Representação adicional de vaguesa pelo uso de advérbios (Hedges)

No diálogo entre as pessoas, é comum a utilização de elementos enfáticos, para reforçar os significados de suas declarações, e elas o fazem através do uso de advérbios. Fulano é *muito* alto, sicrano é *um pouco* alto, etc. A

teoria dos conjuntos difusos, contém recursos que podem ser usados para captar a ênfase provocada pelo uso de advérbios. São os **Hedges**.

Um Hedges modifica matematicamente um conjunto difuso já existente, de forma a incorporar ao seu significado o efeito provocado pelo uso de um advérbio.

Para ilustrar este efeito, pode-se considerar um exemplo. Uma pessoa de 1,80 m pode ser considerada uma pessoa alta em um conjunto de pessoas altas com 0,7 de pertinência. Entretanto, este mesmo indivíduo pode ser considerado muito alto, com pertinência 0,49, se for aplicado um Hedges de concentração (muito) sobre sua situação de alto neste caso:

$$\mu_{\text{con}(A)}(x) = (\mu_A(x))^2$$

A aplicação da potência $n=2$, no exemplo, diminui o valor da pertinência dos elementos que têm valores de pertinência pequenos, aumenta portanto o rigor do conceito expresso pelo conjunto difuso original. Existem outros tipos de Hedges. Como o de dilatação:

$$\mu_{\text{dilat}(A)}(x) = (\mu_A(x))^{0,5}$$

que diminui a restrição do conceito. E outros [DURK1994].

2.7 Operadores difusos

Essa seção é baseada em [DURK1994], que trata apenas dos operadores padrão da teoria dos conjuntos difusos, então as referências a operadores difusos aqui devem ser consideradas relativamente aos operadores

padrão. Nos capítulos posteriores deste trabalho no entanto foi usado o produto algébrico como operador de intersecção.

2.7.1 Intersecção

Na teoria clássica dos conjuntos, a intersecção entre dois conjuntos, contém aqueles elementos que são comuns a ambos. Na teoria dos conjuntos difusos entretanto, o elemento pode pertencer parcialmente aos dois conjuntos, ainda que não pertença completamente a nenhum deles. Assim, quando é considerada a intersecção desses conjuntos, não se pode dizer que um elemento possa pertencer mais ao conjunto da intersecção do que a qualquer um dos conjuntos originais. De acordo com isto, o operador difuso usado para gerar a intersecção entre dois conjuntos difusos A e B definidos em X é dada por:

$$\begin{aligned}\mu_{A \wedge B}(X) &= \min(\mu_A(x), \mu_B(x)) \quad \text{para todo } x \in X \\ &= \mu_A(x) \wedge \mu_B(x) \\ &= \mu_A(x) \cap \mu_B(x)\end{aligned}$$

O símbolo \wedge (“E” lógico) é usado para representar o operador “min” que simplesmente toma o mínimo entre os valores em consideração. Exemplo:

$$ALTO = (0/5, 0,2/5,5, 0,5/6, 0,8/6,5, 1/7)$$

$$BAIXO = (1/5, 0,8/5,5, 0,5/6, 0,2/6,5, 0/7)$$

$$\mu_{ALTO} \wedge \mu_{BAIXO}(x) = (0/5, 0,2/5,5, 0,5/6, 0,2/6,5, 0/7)$$

A intersecção entre os conjuntos dados por ALTO e BAIXO, dada por $\mu_{ALTO} \wedge \mu_{BAIXO}(x)$, pode ser entendida como as pessoas de média estatura, por exemplo, e usada como um novo conjunto difuso.

2.7.2 União

Outra forma de combinar conjuntos difusos é através de sua união. A união de dois conjuntos é compreendida como sendo o conjunto dos elementos que pertencem a pelo menos um deles (ou a ambos). Assim sendo os elementos do conjunto união não podem possuir valor de pertinência menor que o que possuía em qualquer um dos conjuntos originais. A forma padrão que a lógica difusa usa para obter a intersecção entre dois conjuntos é a seguinte:

$$\begin{aligned}\mu_{A \vee B}(x) &= \max(\mu_A(x), \mu_B(x)) \quad \text{para todo } x \in X \\ &= \mu_A(x) \vee \mu_B(x) \\ &= \mu_A(x) \cup \mu_B(x)\end{aligned}$$

O símbolo \vee (chamado “ou” lógico) é usado na lógica difusa para representar a operação “max”, que toma o valor máximo dentre os valores em consideração. Exemplo:

$$\text{ALTO} = (0/5, 0,2/5,5, 0,5/6, 0,8/6,5, 1/7)$$

$$\text{BAIXO} = (1/5, 0,8/5,5, 0,5/6, 0,2/6,5, 0/7)$$

$$\mu_{\text{ALTO} \vee \text{BAIXO}}(x) = (1/5, 0,8/5,5, 0,5/6, 0,8/6,5, 1/7)$$

A união entre os conjuntos dados por ALTO e BAIXO, dada por $\mu_{\text{ALTO} \vee \text{BAIXO}}(x)$, pode ser entendida como as pessoas que não são de estatura média, por exemplo, e também usada como um novo conjunto difuso.

2.7.3 Complementação

Dado um conjunto difuso A , é possível encontrar o seu complemento $\sim A$ através da seguinte expressão:

$$\mu_{\sim A}(x) = 1 - \mu_A(x)$$

Por exemplo, dado o conjunto das *peessoas altas* usado nos exemplos anteriores, este operador pode ser usado para criar o conjunto das *peessoas não altas* ou *peessoas medias ou baixas*.

$$\mu_A(x) = \text{ALTO} = (0/5, 0,2/5,5, 0,5/6, 0,8/6,5, 1/7);$$

$$\mu_{\sim A}(x) = \text{NÃO ALTO} = (1/5, 0,8/5,5, 0,5/6, 0,2/6,5, 0/7).$$

2.8 Conjuntos difusos e FAN

Em várias fases no desenvolvimento dos conceitos de FAN e das unidades FAN, são utilizadas ferramentas da teoria dos conjuntos difusos. Em seguida são mostradas brevemente as principais conexões entre as teoria encontradas neste trabalho a discussão aprofundada, encontra-se nos capítulos subseqüentes.

O primeiro encontro entre FAN e conjuntos difusos ocorreu pela necessidade de representar-se o conceito de um número próximo de um valor dado. O valor de uma característica apresentada para FAN, não é apenas o valor real numérico desta característica, mas um conjunto difuso discreto composto pelos valores próximos a este valor numérico. Para construir estes conjuntos é preciso que se defina uma regra de formação para eles. Esta regra é obtida através do uso de uma função de pertinência estabelecida para tal. Durante os testes nas aplicações, várias funções foram testadas para serem usadas como regra de formação. A função triangular foi a escolhida por proporcionar um bom desempenho computacional sem perdas no que diz respeito à qualidade nos resultados. Ainda durante os testes foram aplicados às funções de pertinência

vários Hedges, que em alguns casos melhoraram os resultados obtidos, e que foram mantidos os que levaram aos melhores resultados em cada aplicação.

Ao gerar o espaço de combinação das características, foi aplicado o operador de intersecção difuso. O operador padrão “min” foi testado, mas o produto algébrico foi o operador de intersecção que levou aos melhores resultados nos testes e foi mantido em todas as aplicações.

O operador de complemento difuso foi utilizado para o calculo dos graus de pertinência de um padrão difuso de entrada e cada unidade FAN (como mostrado nos capítulos futuros).

2.9 Conclusão

Neste capítulo foram apresentados os principais aspectos da teoria dos conjuntos difusos e da lógica difusa: histórico, notação, conceitos e ferramentas. Também foram apresentadas as conexões entre estes aspectos com o desenvolvimento de FAN. Algumas considerações mais específicas no que diz respeito ao reconhecimento de padrões difusos serão feitas no capítulo seguinte que trata especificamente de Reconhecimento de Padrões.

3 Reconhecimento de padrões

3.1 Introdução

A inteligência computacional é uma das tecnologia mais emergentes nestes anos da década de 1990. Um dos componentes que freqüentemente é necessário no desenvolvimento de sistemas inteligentes é o reconhecimento de padrões. Falando de grosso modo, o reconhecimento de padrões (RP), é a ciência que trata da descrição ou classificação (reconhecimento) de medidas. Não há dúvidas que o RP se constitui em uma tecnologia útil e importante, que vem se desenvolvendo rapidamente com interesses e participações interdisciplinares. RP não se reduz a uma forma de aplicação mas apresenta um leque de abordagens possíveis que podem ou não ser relacionadas entre si.(baseado em [SCHALKOFF1992]).

3.2 Definição de Reconhecimento de Padrões (RP)

Reconhecimento de padrões pode ser definido como um processo de identificar estruturas nos dados por comparações com estruturas conhecidas; as estruturas conhecidas são desenvolvidas através de métodos de classificação em [ROSS1995].

Ainda se encontra a seguinte definição em Klir [KLIR1995]: “O reconhecimento de padrões pode ser definido como um processo pelo qual buscam-se estruturas nos dados e classificam-se estas estruturas dentro de

categorias tais que o grau de associação é maior entre as estruturas da mesma categoria e menor entre as categorias de estruturas diferentes. As categorias relevantes, são usualmente caracterizadas por estruturas prototípicas derivadas da experiência do passado. Cada categoria pode ser caracterizada por mais de uma estrutura prototípica”.

3.3 Estrutura típica de um sistema de reconhecimento de padrões

A estrutura de um sistema típico de reconhecimento de padrões é mostrada na figura 3.1. Note que ela consiste em um sensor (uma câmera por exemplo), um mecanismo de extração de características (algoritmo), e um algoritmo de descrição ou classificação (dependendo da aplicação). Complementarmente, é usual que alguns dados que já tenham sido classificados ou descritos estejam disponíveis para treinar o sistema (este conjunto de dados é chamado conjunto de treinamento). Extraído de [SCHALKOFF1992].

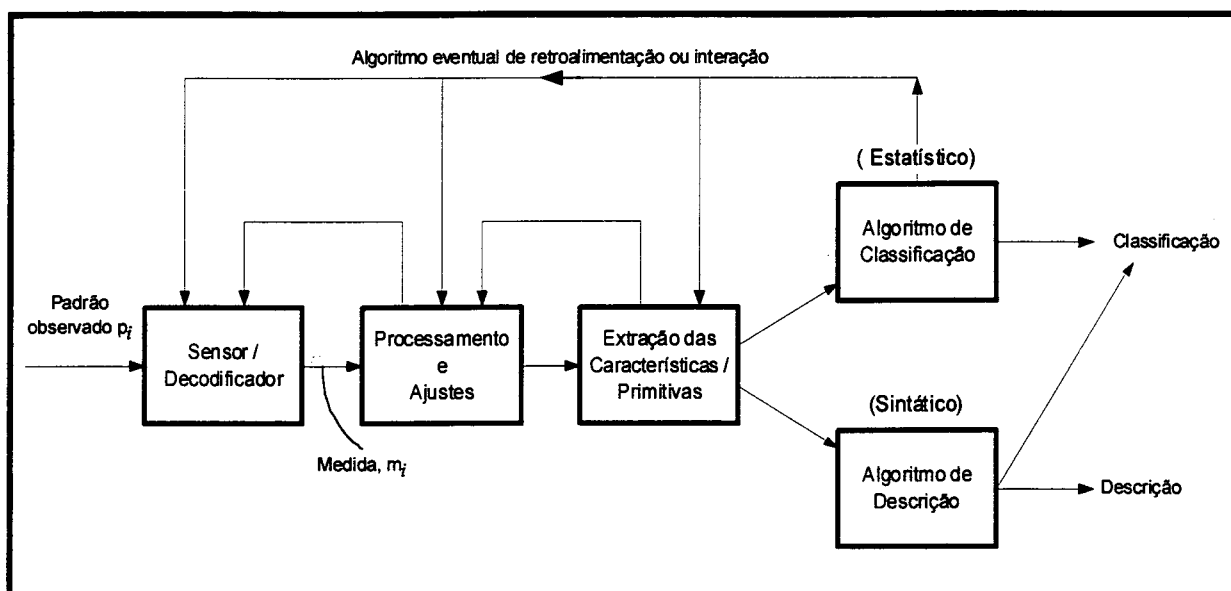


Figura 3.1 – Sistema típico de reconhecimento de padrões

3.4 Aplicações do Reconhecimento de Padrões (RP)

O termo reconhecimento de padrões envolve uma gama enorme de problemas de processamento de informação, com grande significância prática,

desde o reconhecimento de voz e de caracteres feitos manualmente, até à detecção de erros em equipamentos ou diagnósticos médicos. Não é possível comentar todas as áreas onde seria útil e por vezes mesmo necessário aplicar RP. Tal comentário se justifica se for levado em conta que as pessoas aplicam seus próprios métodos de reconhecimento de padrões em praticamente todas as áreas da atividade humana. A racionalidade pura, não é suficiente para o bom desempenho na resolução da maioria dos problemas. Tome-se como exemplo o jogador de xadrez. A análise de todas as possibilidades em uma dada posição complexa é virtualmente impossível para um jogador, que no entanto se for hábil, desenvolve suas técnicas para identificar os padrões das várias posições as quais se deparou em suas experiências, e usar estas técnicas para obter posições vantajosas em seus jogos. Na prática computacional, pode-se enumerar algumas áreas (algumas áreas) onde vem sendo usado RP:

- processamento de imagens;
- visão computacional;
- classificação/análise dos sinais de radar;
- reconhecimento de fisionomia;
- reconhecimento de voz;
- tratamento de impressões digitais;
- reconhecimento de caracteres;
- análise de manuscritos;
- análise de eletrocardiogramas;
- diagnósticos médicos.

Nesta dissertação, FAN é aplicado no problemas da Íris de Fisher [FISHER1936], no diagnóstico de saúde financeira [PACHECO1996] e [MARTINS96], e no reconhecimento de cromossomas [TODESCO95]. O primeiro caso é normalmente usado para finalidades didáticas para testar várias técnicas de RP. Os dois outros foram tratados nos trabalhos citados acima usando abordagens de redes neuronais artificiais.

3.5 Abordagens em RP

3.5.1 Introdução

São diversas as abordagens possíveis em RP, como foi comentado na introdução deste capítulo. Este trabalho mesmo objetiva apresentar uma forma alternativa para tratar o assunto. Duas destas abordagens no entanto estão presentes em praticamente toda a literatura que versa sobre o assunto, podendo ser consideradas as abordagens clássicas de RP. São o RP estatístico e o RP sintático os quais são tratados nas seções 3.4.2 e 3.4.3. Por serem bastante relacionados com o RP através de FAN, acrescentaram-se as seções 3.4.3 e 3.4.4 que tratam respectivamente do RP usando redes neuronais e RP difuso.

3.5.2 Reconhecimento de padrões estatístico

Segundo Bishop em [BISCHOP1995], a forma mais geral e natural de formular soluções para o reconhecimento de padrões é o RP estatístico, através do qual é reconhecida a natureza estatística tanto da informação que se quer representar quanto dos resultados que devem ser expressados. O reconhecimento de padrões estatístico é uma área de conhecimento bem estabelecida e possui uma longa história.

Robert Schalkoff [SCHALKOFF1992], diz que, o reconhecimento de padrões estatístico como sugere o próprio nome, assume uma base estatística para os algoritmos de classificação. Um conjunto de medidas, que denotam as características, é extraído dos dados de entrada e usado para associar cada vetor de características a uma de entre c classes. Presume-se que as características são

geradas por um estado da natureza, e portanto existe um modelo subordinado a um estado com um conjunto de probabilidades e/ou funções de densidade de probabilidade correspondente passível de ser usado para representá-lo.

3.5.3 Reconhecimento de padrões sintático

Muitas vezes, as informações significativas em um padrão, não consistem apenas na presença ou ausência de valores numéricos de um conjunto de características. Ao invés disto, a inter-relação ou interconexão das características produzem informações estruturais importantes, que facilitam a descrição ou classificação. Esta é a base do RP sintático. Portanto é necessário ao se usar a abordagem de RP sintático, quantificar e extrair as informações estruturais importantes para poder avaliar a similaridade estrutural entre os padrões.

Tipicamente, o RP sintático formula uma descrição hierárquica de padrões complexos, construída a partir de sub-padrões mais simples, sendo que no nível mais baixo, se encontram os elementos mais simples, extraídos dos dados de entrada que são chamados de primitivas.

3.5.4 Reconhecimento de padrões por redes neurais

Alguns trabalhos como é o caso encontrado em Bishop [BISCHOP1995], adotam o ponto de vista que a abordagem de RP através do uso de redes neurais, é uma extensão do RP estatístico convencional, sendo a segunda forma considerada como um aperfeiçoamento da primeira.

Uma rede neuronal ou connexionista é definida em Cichoki & Unbehauen [CICHOKI&UNBEHAUEN1993], como um sistema de processamento de sinal ou informação, composto por um grande número de elementos simples de processamento, chamados neurônios artificiais, ou simplesmente nós, que são interconectados por elos diretos chamados de conexões, que colaboram para realizar um processamento paralelo distribuído (PPD) para realizar uma tarefa computacional desejada.

A tarefa computacional citada no parágrafo anterior pode ser a de que: dado um vetor de características deve-se associa-lo a uma classe em um determinado problema (i. e. reconhecimento de padrões). Na maioria das aplicações de RP usando-se redes neuronais são estabelecidas conexões entre os valores de todas as características que definem os padrões com camadas intermediárias de neurônios e as todas as classes objetivo que são consideradas como a camada de saída. O treinamento da rede é realizado corrigindo os pesos nas conexões para estabelecer-se as relações entre as características e classes que promovam a melhor discriminação possível entre os padrões de classe diferentes. Assim ao ser apresentado à rede um novo padrão, ela indicará a classe que o represente melhor na camada de saída (aqui é importante salientar que o conceito de neurônio usado nas redes connexionistas é diferente daquele que dado às unidades FAN. Nas redes, um peso ou conexão recebe a denominação de neurônio, enquanto um neurônio livre associativo é constituído de uma série de pesos ou parâmetros reunidos para um propósito comum, no caso de RP, representar uma única classe). Existem vários trabalhos publicados que tratam exclusivamente do reconhecimento de padrões utilizando-se as redes neuronais como o de Bishop [BISCHOP1995], que já citamos. O trabalho de dissertação de mestrado de Dandolini [DANDOLINI1997], contém um estudo detalhado de diversos tipos de redes neuronais aplicadas ao RP além de uma comparação delas

com FAN para um mesmo problema. As redes cujos resultados obtidos em RP para as aplicações específicas que são apresentadas neste trabalho foram MLP (Multi-Layer Perceptron) com Backpropagation e RBF (Radial Basis Function), que podem ser encontradas nas referências citadas acima.

3.5.5 Reconhecimento de padrões usando conjuntos difusos

A utilidade da teoria dos conjuntos difusos no reconhecimento de padrões já foi reconhecida em meados da década de 1960, e hoje a literatura a respeito de RP difuso é bastante extensa segundo Klir [KLIR1995]. Existem portanto muitas maneiras de se aplicar a teoria dos conjuntos difusos em RP. Uma delas é proposta neste trabalho, porém existe duas formas clássicas de RP difuso que vale comentar. São: os métodos de reconhecimento de padrões através de listas de pertinências, e os métodos difusos sintáticos.

No método clássico da lista de pertinência, cada classe é caracterizada por um conjunto de padrões que é armazenado no sistema de reconhecimento de padrões. Um padrão desconhecido a ser classificado é comparado com os padrões armazenados um a um. O padrão é classificado como um membro de uma classe se ele combinar com um dos padrões pertencentes a esta classe.

No método sintático difuso, um padrão é representado por uma cadeia de sub-padrões concatenados chamados de primitivas. Estas primitivas são vistas como o alfabeto de uma linguagem formal. Um padrão então, é uma sentença gerada por alguma gramática. Todos os padrões, cujas sentenças são geradas pela mesma gramática, pertencem a uma mesma classe. Um padrão

desconhecido então, é classificado como pertencente a uma classe particular, se ele puder ser gerado pela gramática correspondente a ela. (de [KLIR1995]).

4 Modelo proposto

4.1 Introdução

Uma palavra, “complexidade” descreve os principais obstáculos no que se refere à emulação de inteligência. Ela pode ser observada na incerteza, informações incompletas, relacionamentos dinâmicos, comportamento caótico, etc.. Os problemas mais significativos resolvidos pelas pessoas possuem alguma ou todas estas características. Conforme ocorre o avanço na inteligência computacional, torna-se claro que o uso de métodos isolados é capaz de modelar apenas uma parte daquilo que caracteriza a inteligência. Nos últimos anos, vários modelos híbridos [KANDEL1992] [MEDSKER1995] [GOONATILAKE1995] (em [RAITTZ1997]) foram desenvolvidos combinando qualidades de diferentes tecnologias para simular comportamentos inteligentes mais acuradamente.

Há duas formas de combinar métodos inteligentes: (a) melhorando um paradigma através de inclusão de outro; ou (b) desenvolvendo novos modelos híbridos compostos por uma combinação de diferentes técnicas. *Free Associative Neurons* (FAN) é um exemplo do segundo caso. A base de FAN são as redes connexionistas, a modelagem difusa e representação de padrões. FAN é baseado na noção do aprendizado por adaptação neuronal, utiliza técnicas de conjuntos difusos para promover a granularidade de informações, e permite incluir diferentes métodos de associação de padrões para melhorar a sua capacidade de aprendizado.

A motivação para o desenvolvimento de FAN foi elaborar um método capaz de simular abstrações. Devido a esta capacidade cognitiva, os seres humanos compreendem um cenário completo baseado apenas nas partes que o compõem. De fato, na linguagem natural as pessoas resumem a informação, classificando objetos em estruturas de categorias similares (palavras), evitando a sobrecarga de dados. Em FAN, a noção de classificação de padrões é feita por três processos: (a) expansão do espaço de entrada; (b) projeção da vizinhança do padrão resultante; (c) quantificação e aprendizado do grau de similaridade entre as classes FAN e as projeções dos padrões.

Neste capítulo, baseado em [RAITZ1997] são discutidos os principais componentes de FAN. Começa por trazer uma visão geral do método e seus princípios seguido-se uma discussão sobre modelagem de padrões e processos de aprendizados usados em FAN. Na conclusão é apresentado um resumo dos principais aspectos abordados no capítulo.

4.2 Problema

O Problema ao qual é aplicado o FAN consiste na associação de padrões à uma determinada classe.

O objetivo é que sendo dado um padrão x qualquer (padrão de entrada) em um determinado domínio de problema, seja possível encontrar a classe à qual x possa ser associado mais adequadamente. Ver capítulo de reconhecimento de padrões.

4.3 Modelo proposto (FAN)

Em FAN, cada padrão de entrada é expandido em uma vizinhança difusa. Cada conjunto suporte desta vizinhança é a combinação dos valores das características próximos dos originais. O grau de similaridade entre a vizinhança difusa e o padrão original de entrada é feita através das técnicas utilizadas na teoria dos conjuntos difusos [KLIR1995]. O aprendizado acontece através da projeção da vizinhança difusa no espaço FAN. Há uma unidade FAN para cada classe do domínio do problema. Cada unidade é uma matriz composta por todas as combinações de características observadas em sua classe correspondente. Durante o treinamento, cada combinação é representada por uma célula difusa que contém um peso correspondente à sua frequência de ocorrência e grau de pertinência. O treinamento é baseado no *reforço* nas células projetadas de FAN (se a classificação foi correta) ou em *esquecimento* (se houve uma classificação incorreta).

4.3.1 Modelagem dos dados

Seja $\mathbf{x} = (x_1, \dots, x_p) \in \mathbb{R}^p$ um padrão, descrito por p características. Cada valor de x_i ($1 \leq i \leq p$) representa a avaliação no espaço i -dimensional das características, ou simplesmente, o valor da i -ésima característica. A tarefa de aprendizado pode ser definida como a busca pelo melhor modelo para relacionar os padrões de entrada com os valores das funções discriminantes (valores de saída). Onde, “melhor” significa a aproximação ótima entre o modelo e a respectiva resposta. A modelagem dos dados procura representar as características dos padrões de forma a contribuir com a metodologia do

aprendizado. Em FAN, dois princípios básicos fundamentam o processo de representação dos dados:

- um padrão representa mais que um simples ponto em termos de informação. Assim na modelagem será considerada uma região ao redor do padrão;
- cada padrão contém mais informações, que vão além dos valores individuais. Fazendo a correlação dos valores das características, a performance pode ser aumentada.

O primeiro aspecto está relacionado com a *decomposição do padrão*, ou seja, a geração de uma vizinhança em torno do padrão de entrada. A vizinhança consiste em um conjunto de padrões difusos no espaço de entrada, próximos ao padrão original. O outro passo no processo de modelagem dos dados é a *projeção de cada vizinhança difusa* em uma região que denomina-se espaço FAN, que consiste em um conjunto de pontos do espaço \mathcal{R}^{2H} que é comentado detalhadamente mais adiante. O algoritmo FAN mede a similaridade entre estas projeções e as unidades FAN, como será descrito posteriormente. Em suma, o processo de modelagem é realizado pelos seguintes procedimentos:

- decomposição do padrão (vizinhança difusa);
- projeção da vizinhança difusa.

Na decomposição, o padrão de entrada é transformado em um conjunto de padrões difusos. Dado um padrão de entrada \mathbf{x} , a seguinte expressão representa o conjunto de padrões difusos:

$$\bar{\mathbf{x}}_z = (\mu(v_1(l_1))/v_1(l_1), \dots, \mu(v_p(l_p))/v_p(l_p)) \quad (4.1)$$

onde:

$$z = 1, (2D + 1)^p;$$

D é o *raio* de decomposição. Ele estabelece o número de inteiros em cada lado da coordenada original x_i (*centro*) do vetor de entrada \mathbf{x} ;

$v_i(l_i)$ é o valor da i -ésima coordenada inteira, próxima ao valor original x_i . O índice $l_i = 1, 2D+1$, variando para a vizinhança de cada coordenada ;

$\mu_{x_i}(v_i(l_i))$ é a função de pertinência que mede a similaridade entre o valor $v_i(l_i)$ e o original x_i ($i=1,p$).

Um exemplo ilustra melhor a Expressão (4.1). Suponha que o vetor de entrada $\mathbf{x} = (20, 70.5, 90)^1$; primeiramente, para as coordenadas x_1, x_2 e x_3 , o processo de decomposição do padrão gera p conjuntos difusos, cada um de tamanho $2D+1$. Supondo $D = 3$, os números difusos em cada característica são:

- $\mathbf{v}_1(x'_1) = (0.25/17, 0.50/18, 0.75/19, 1.00/20, 0.75/21, 0.50/22, 0.25/23);$
- $\mathbf{v}_2(x'_2) = (0.25/67, 0.4/68, 0.7/69, 0.9/70, 1.0/70.5, 0.86/71, 0.57/72, 0.3/73);$
- $\mathbf{v}_3(x'_3) = (0.25/87, 0.50/88, 0.75/89, 1.00/90, 0.75/91, 0.50/92, 0.25/93).$

onde 20, 70.5 e 90 (*i.e.*, os valores originais das características) são denominados os *centros* dos conjuntos suportes de \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 , respectivamente. O próximo passo consiste em gerar as vizinhanças difusas $\bar{\mathbf{x}}_i$ aplicando a expressão (4.1). Os valores das coordenadas \bar{x}_i são tomados dos vetores discretizados \mathbf{v}_i . O número total de vizinhos é então, o Produto Cartesiano dos conjuntos suportes das características difusas com $(2D+1)^p$ pontos, conforme indicado em (4.1). Desta forma, o primeiro vizinho leva os primeiros valores de características no qual \mathbf{v}_i : $\bar{\mathbf{x}}_1 = (0.25/17, 0.25/67, 0.25/87)$. Este vetor poderia ser visto como o vizinho menos similar do lado esquerdo do vetor de entrada original $\mathbf{x} = (20, 70.5, 90)$. Outros exemplos seriam:

- $\bar{\mathbf{x}}_2 = (0.25/17, 0.25/67, 0.50/88);$
- $\bar{\mathbf{x}}_3 = (0.25/17, 0.25/67, 0.75/89);$
- $\bar{\mathbf{x}}_{\dots} = (0.50/18, 0.57/72, 0.75/91).$

Uma vez criados os vizinhos difusos, o processo de modelagem dos dados dá início à projeção da vizinhança. A idéia é projetar o padrão original \mathbf{x}

¹ Em problemas práticos esses números deveriam ser definidos tais que o valor do parâmetro D possa ser o mesmo para todas as características.

sobre o espaço de combinações das características. Sua dimensão é igual ao produto cartesiano difuso da vizinhança das características, onde a similaridade em relação aos valores originais é obtida através de técnicas difusas. Isso é feito projetando cada vizinho difuso \bar{x}_z sobre o espaço cartesiano difuso, gerando a projeção F_z .

O primeiro parâmetro a ser ajustado é o *nível de combinação das características*, denominado H . Este fator tem impacto direto na dimensão do espaço FAN. Quanto maior for H , maior se torna a dimensão do espaço FAN. O escopo de valores de H é o intervalo $[1, p]$, onde p é a dimensão do espaço de entrada. Dado um vetor difuso de vizinhança como em (1), há p características a serem combinadas em H -uplas. O conjunto destas combinações formam a projeção. Desta forma, um fator $H = 2$ implica que as características serão combinadas em pares. A projeção completa F_z é formada por todas as $C_p^H + p$ combinações de características. Esta combinação é composta pelos índices dos vetores \bar{x}_z e pelos valores das coordenadas $v_j(i_j)$ a serem combinados. Cada combinação é avaliada pelo produto dos graus de pertinência correspondentes $\mu_{x_j}(v_j(i_j))$. A projeção de F_z é determinada então por:

$$F_z \in \mathbf{R}^{m(C_p^H + p)} = \text{Proj}(\bar{x}_z) = (f_m^z, \mathbf{a}, \boldsymbol{\alpha}) \quad (4.2)$$

onde $\mathbf{a} = (a_1, \dots, a_H)$ é o vetor dos índices, com $1 \leq a_j \leq p$, $\boldsymbol{\alpha} = (v_1(i_1), \dots, v_H(i_H))$ e

$$f_m^z = \prod_{j=1, H} \mu(v_j(a_j)), \text{ com } m = 1, (C_p^H + p). \quad (4.3)$$

A projeção de um vizinho difuso pode ser melhor compreendida através de um exemplo. Tomando o vetor difuso $\bar{x}_z = (0.25/17, 0.40/68, 0.25/87)$ discutido anteriormente, a projeção difusa F_1 com um fator $H=2$ é dada pela Tabela 4.1.

m	i	j	$\mu_{x_i}(V_i(l_i))$	$\mu_{x_j}(V_j(l_j))$	V_i	V_j	f_m^z
1	1	1	0.25	0.25	17	17	0.0625
2	1	2	0.25	0.40	17	68	0.1000
3	1	3	0.25	0.25	17	87	0.0625
4	2	2	0.40	0.40	68	68	0.1600
5	2	3	0.40	0.25	68	87	0.1000
6	3	3	0.25	0.25	87	87	0.0625

Tabela 4.1 - Exemplo da projeção de um vizinho

4.4 Método de Aprendizado FAN

As Expressões (4.2) e (4.3) descrevem como um vizinho difuso é representado em um subespaço do FAN. Na implementação do aprendizado entretanto, deve-se considerar não apenas um vizinho. Na expressão abaixo, F é calculado usando-se todos os vizinhos possíveis para H e D definidos para a aplicação. Em alguns casos práticos não é usada a totalidade, mas sim uma quantidade que seja representativa, evitando problemas de explosão combinatorial. O número destas combinações é usado como parâmetro nas aplicações e recebem o nome de *número de combinações das características*. A *projeção da vizinhança difusa* é a união dos vizinhos difusos projetados de F_z :

$$F = \bigcup_{z=1, (2D+1)^P} F_z. \quad (4.4)$$

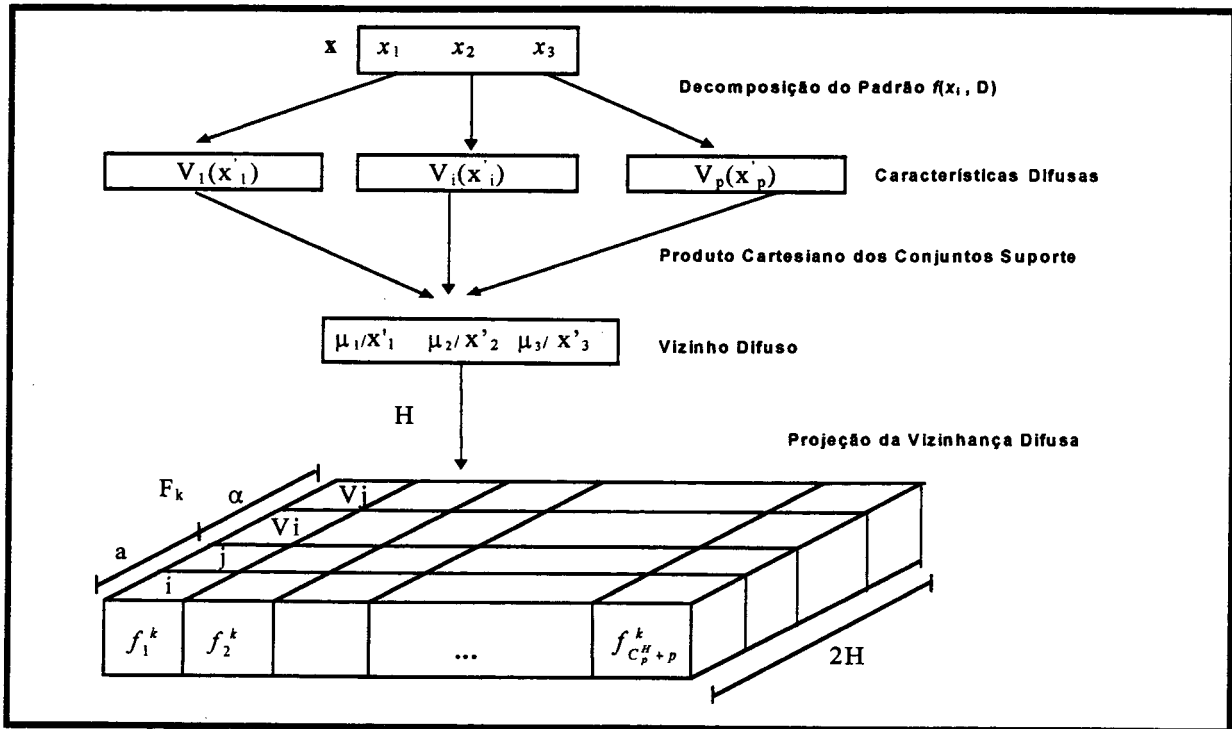


Figura 4.1 – Visão esquemática do processo de modelagem de dados.

A projeção completa da vizinhança difusa F , é um conjunto de $(2D+1)^p$ cubos similares àqueles, na parte inferior da Figura 4.1. A Figura 4.2 representa a união mostrando cada elemento F_z do conjunto resultante F .

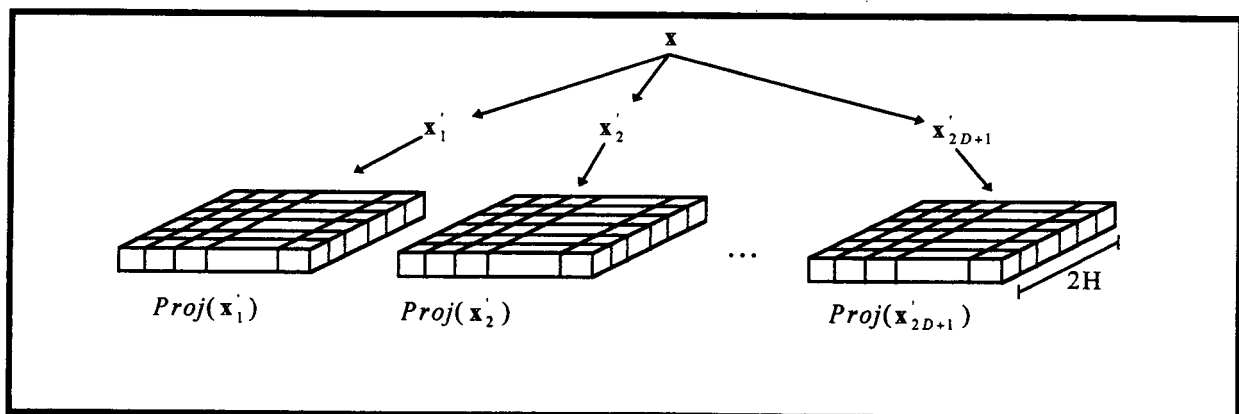


Figura 4.2 – Projeção da vizinhança difusa.

O aprendizado envolve a comparação entre a projeção F e todas as unidades FAN^j ($j=1, C$, onde C é o número de classes do domínio do problema).

A dimensão de F é definida pela expansão do padrão de entrada x (*i.e.*, depende dos valores das coordenadas x_i de x e dos parâmetros D, H). FAN^j entretanto possui uma dimensão maior, incluindo todas as combinações de características que ocorrem para todo $x^k \in FAN^j$. A comparação é feita avaliando o grau ao qual F é um subconjunto de FAN^j .

O cálculo da similaridade entre F e FAN^j , bem como as expressões usadas para reforço e esquecimento não representam a essência do método proposto, podendo variar entre uma aplicação e outra. As expressões usadas na sequência, bem como no algoritmo apresentado neste capítulo são as mesmas utilizadas nas aplicações do capítulo seguinte.

A similaridade entre F e FAN^j é dada por:

$$S(F, FAN^j) = 1 - \prod_g \left(1 - \frac{f_g \cdot fan_g^j}{fan^j} \right) \quad (4.5)$$

onde fan_g^j é uma célula do conjunto difuso FAN^j , f_g é uma célula do conjunto difuso F (obtido pela Expressão (3)), g é o índice que cobre todo o domínio de FAN^j que possui um correspondente em F e fan^j é o número total de ocorrências na unidade FAN^j .

A Expressão (5) mede a similaridade da projeção F em cada unidade FAN, FAN^j .

A maior similaridade indica a classe associada ao padrão de entrada x . Em outras palavras, a saída da rede é j^* -ésima classe, onde $j^* \in [1, C]$ é o

índice da unidade FAN que apresenta maior similaridade, FAN^j . Isto é obtido por:

$$S(F, FAN^{j*}) = \max_j S(F, FAN^j) \quad (4.6)$$

Uma vez que a saída da rede é determinada, acontece o aprendizado (*backward computation*). Primeiramente ocorre a checagem do acerto na classificação ($j^* = j^t$, onde j^t é a classe do domínio associada ao padrão de entrada x) ou erro de classificação (caso contrário). Em ambos os casos a mudança dos valores de pertinência de FAN^{j*} é determinada pelo grau de pertinência da célula f_g .

Quando a saída da rede for correta, o procedimento consiste no fortalecimento das células FAN^{j*} interceptadas pela projeção F . O fortalecimento consiste em somar ao conteúdo atual da célula fan_g^j o valor de pertinência f_g . A pertinência será determinada pela somatória das pertinências na célula dividida pelo total de ocorrências na unidade: $\frac{fan_g^j}{fan^j}$.

No caso de ocorrer uma classificação errada, o algoritmo realiza um procedimento de *esquecimento*, diminuindo o valor das células FAN^{j*} interceptadas pela projeção F . neste caso a pertinência f_g é subtraída do conteúdo da célula, se este conteúdo tornar-se negativo após a subtração, seu valor será definido como sendo zero.

A Figura 4.3 representa a idéia que se encontra por trás dos procedimentos de medidas de similaridade, *reforço* e *esquecimento*. Quanto mais escuras forem as células maior é o grau de pertinência em relação ao conjunto difuso (F ou FAN^{j*}).

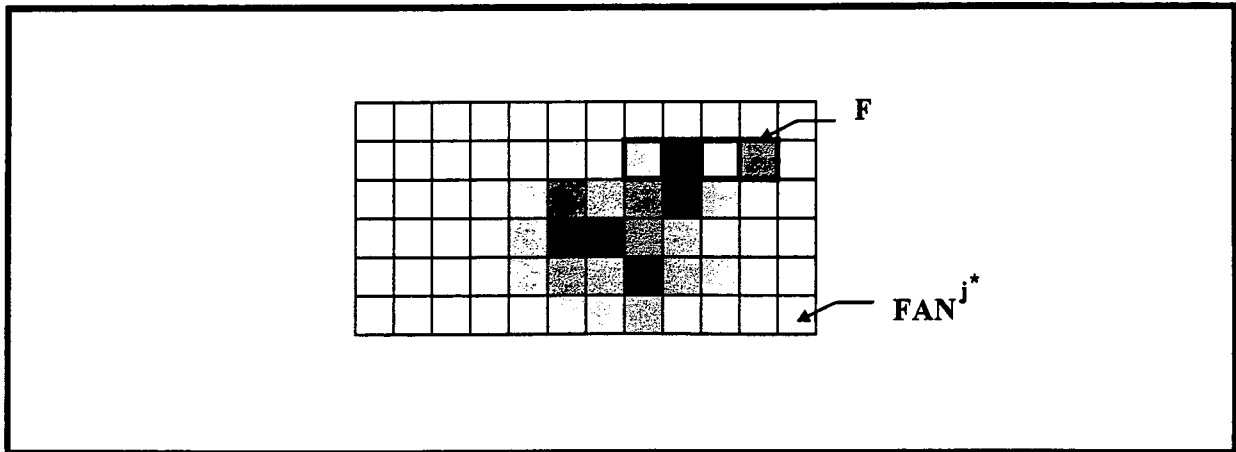


Figura 4.3 – Intersecção Entre F e FAN^j *

4.5 Algoritmo FAN

Conjunto de Treinamento: $P_k(\mathbf{x}^k, \mathbf{t}^k)$, ($k = 1, N$), tal que $\mathbf{x}^k \in \mathbb{R}^p$ e $\mathbf{t}^k \in \mathbb{R}^q$.

onde: \mathbf{x}^k é o k -ésimo padrão a ser aprendido; \mathbf{t}^k indica a classe à qual \mathbf{x}^k pertence; \mathbb{R}^p é o espaço das características dos padrões; \mathbb{R}^q é o espaço dos padrões e N é o número de dados de treinamento.

inicialização:

faça $FAN^j = \mathbf{0}$ para $j = 1, C$, onde C é o número de classes no domínio do problema.

ajuste o nível de combinações de características H e o raio de decomposição D , tal que $1 \leq H \leq p$, e $D > 0$.

Para cada par $P_k(\mathbf{x}^k, t^k)$ faça o seguinte:

primeiro passo: decomposição do padrão.

Para cada coordenada x_i^k do vetor \mathbf{x}^k gere as seqüências $V_i^k = [\text{int}(x_i^k - D), \dots, \text{int}(x_i^k), \dots, \text{int}(x_i^k + D)]$, para $i=1, p$. Escolha uma função de pertinência $\mu_{x_i^k}(x)$ tal que $\mu_{x_i^k}(x < x_i^k - D) = 0$, $\mu_{x_i^k}(x_i^k) = 1$, $\mu_{x_i^k}(x > x_i^k + D) = 0$. Crie os vizinhos difuso $\bar{\mathbf{x}}_z^k \in \mathbb{R}^p$, onde $z = 1, (2D+1)^p$ assumindo valores nas seqüências V_i^k como conjunto suporte, i.e., $\bar{\mathbf{x}}_z^k = (\mu_{x_1^k}(v_1^k(l_1))/v_1^k(l_1), \dots, \mu_{x_p^k}(v_p^k(l_p))/v_p^k(l_p))$, onde cada $v_i^k(l_i)$ é um membro da seqüência V_i^k , o qual $1 \leq l_i \leq (2D+1)$, $i=1, p$.

Segundo passo: projeção da vizinhança difusa.

(i) *projeção do vizinho*: Para cada número difuso $\bar{\mathbf{x}}_z^k$, $z = 1, (2D+1)^p$ encontre a projeção $\mathbf{F}_z \in \mathbb{R}^{m(C_p^H + p)} = \text{Proj}(\bar{\mathbf{x}}_z^k) = (f_m^z, \mathbf{a}, \alpha)$; \mathbf{a} é o vetor de índices $\mathbf{a} = (a_1, \dots, a_H)$, sendo $1 \leq a_j \leq p$, $\alpha = (v_1^k(i_1), \dots, v_H^k(i_H))$, $f_m^z = \prod_{j=1, H} \mu_{x_j^k}(v_j^k(a_j))$, com $m = 1, (C_p^H + p)$.

(ii) *projeção da vizinhança*: calcule a projeção da vizinhança difusa \mathbf{F}^k tal que

$$\mathbf{F}^k = \bigcup_{z=1, (2D+1)^p} \mathbf{F}_z.$$

Terceiro passo: grau de similaridade.

$$S(\mathbf{F}, \mathbf{FAN}^j) = 1 - \prod_g \left(1 - \frac{f_g \cdot fan_g^j}{fan^j} \right), \text{ onde } g \text{ é o índice que cobre todo o domínio de FAN,}$$

correspondente ao domínio de \mathbf{F}^k e fan^j é o número total de ocorrências de reforços na unidade \mathbf{FAN}^j .

Quarto passo: saída da Rede.

Encontre \mathbf{FAN}^{j^*} , tal que $S(\mathbf{F}^k, \mathbf{FAN}^{j^*}) = \max_j S(\mathbf{F}^k, \mathbf{FAN}^j)$. Faça j^* ser a saída da rede

Quinto passo: Retroprocessamento.

Escolha o fator $\lambda = f_g$, como parâmetro de reforço e aplique a expressão [7] para FAN^{j^*} . Verifique se $j^* = j^{tk}$, onde j^{tk} é a classe domínio associada ao objetivo t^k . Escolha o fator λ de acordo com a classificação da rede: se a classificação for correta, ou seja, $j^* = j^{tk}$, escolha λ como parâmetro de *reforço*. Se houve uma classificação incorreta, tome $\lambda = -f_g$, como sendo o parâmetro de *esquecimento*.

Em ambos os casos, utiliza a seguinte expressão FAN^j :

$$fan^{NEW z^*}_g = fan^{OLD z^*}_g + \lambda \quad (4.7)$$

4.6 Conclusão

Neste capítulo é introduzida uma nova forma para reconhecimento de padrões. FAN é baseado no aprendizado conexionista e modelagem difusa. Os principais passos para a realização do reconhecimento consistem em: (i) gerar vizinhança do padrão; (ii) gerar as combinações das características que compõem os vizinhos; e (iii) testar a semelhança entre a projeção assim obtida e cada classe do domínio do problema, representada por unidades FAN alterando-as conforme os resultados.

Ainda que tenha sido mostrado um algoritmo com expressões bem definidas e para o espaço completo das projeções, os pressupostos acima podem ser mantidos, alteradas as funções de similaridade e armazenamento, bem como o domínio das projeções pode ser reduzido, tomando-se, ao invés de todos os

vizinhos e combinações de H , apenas uma parte representativa da mesma. Isto foi feito na maior parte das aplicações práticas.

Os resultados obtidos por **FAN** são comparáveis aos dos métodos tradicionais de redes neuronais, em termos de classificação e superior em termos de estabilidade no aprendizado e capacidade de generalização, como é demonstrado no próximo capítulo.

5 Aplicações

5.1 Introdução

O principal objetivo ao se realizar as aplicações de FAN e testar seu desempenho, é demonstrar como a técnica evoluiu em suas diversas fases. Não é possível desenvolver um método prático de reconhecimento de padrões sem relacioná-lo a uma aplicação.

No período entre novembro de 1996 e outubro de 1997, foi estudado o desempenho do modelo proposto em três aplicações. Na primeira fase, o método foi implementado e testado no conhecido problema de discriminação da Íris de Fisher. Apesar de constar freqüentemente na literatura como uma aplicação usada em testes de técnicas de RP, resolver o problema da Íris não possui muito sentido prático, a não ser como trabalho acadêmico. Na segunda fase, FAN foi aplicado na determinação da classe que um cromossoma pertence no grupo de Denver. As informações e os dados utilizados nesta fase foram obtidos na tese de doutorado de Todesco [TODESCO1995]. A terceira fase de aplicação foi realizada no problema de determinação da saúde financeira de pequenas empresas, desenvolvido por Pacheco [PACHECO1996] e Martins [MARTINS1996], em suas teses de doutorado.

As implementações foram realizadas todas em linguagem C em ambiente DOS para os casos onde a taxa de combinação das características foi considerada $H=1$ (a maioria dos testes). Nos casos onde esta taxa foi aumentada para $H=2$ foram feitas aplicações com arquivos de bancos de dados utilizando-se

a linguagem CLIPPER SUMMER 87. Neste segundo caso os testes deverão ser realizados novamente no futuro com implementações em ambiente WINDOWS em trabalhos futuros.

Os parâmetros alterados durante os testes em todos os casos foram:

- o intervalo de discretização;
- o parâmetro de combinações de características H ;
- o número de combinações de características utilizado;
- o raio da função difusa;
- a função de pertinência;
- o número de épocas de treinamento.

A seguir são apresentadas cada uma das fases de aplicação. Na seqüência apresenta-se uma série de experimentos realizados sobre os conjuntos de dados relativos aos cromossomas, que visam analisar o comportamento de FAN variando-se o número de informações consideradas, em função das taxas de erro e a confusão entre a resposta da rede e os resultados reais das observações. Nessa seção, também são avaliadas as curvas das taxas de erro nos conjuntos de teste e treinamento em função do número de épocas realizadas. No final do capítulo são apresentadas as conclusões e comentários.

5.2 Primeira fase - ÍRIS DE FISHER

5.2.1 Definição do problema

O conjunto de dados da Íris, foi popularizado por R. A. Fisher (1936), ao ilustrar os princípios da análise discriminante. Ele contém 6 variáveis com 150 observações [FISHER1936]. As variáveis representam:

- o número que representa o tipo da espécie da flor;
- o nome da espécie (Íris Setosa, Íris Virgínia e Íris Versicolor);
- largura da pétala;
- comprimento da pétala;
- largura da sépala;
- comprimento da sépala.

5.2.2 Comentários

Nesta fase foi construído o primeiro protótipo de FAN. O objetivo foi aplicar o método a um problema simples para avaliar a sua aplicabilidade.

Foram criadas três unidades FAN, uma para representar cada tipo de Íris, com duas características como valores de entrada. O procedimento foi feito interativamente, sendo os padrões apresentados a FAN um a um, quando o algoritmo respondia a classe que julgava que ele representava mais adequadamente. Após isto, tendo ou não a classificação sido correta, a supervisão era feita através da indicação da classe à qual o padrão realmente pertencia para a realização do treinamento.

5.2.3 Resultados e comentários

Nesta fase, apenas uma configuração foi testada obtendo-se excelentes resultado:

- intervalo de discretização com 100 elementos;
- parâmetro de combinações de características $H = 1$;
- número de combinações de características utilizado = 8;
- raio da função difusa = 2;
- função de pertinência triangular;
- número de épocas (não relevante).

Observou-se que após ser apresentado um padrão de cada classe, de diversas formas, o algoritmo indicava com acerto os padrões subseqüentes na grande maioria dos casos, durante todo o processo de treinamento. No teste com uma época de treinamento realizado com os primeiros 10 padrões de cada classe do conjunto de Fisher, FAN obteve 100% de acerto nos 40 padrões restantes de cada classe.

Foram estes resultados positivos que motivaram a continuação dos trabalhos de pesquisa do método e a realização de aplicações em problemas mais complexos.

5.3 Segunda fase - Classificação de cromossomas no grupo de Denver

5.3.1 Definição do problema

Lemos em Todesco [TODESCO1995]: As características mais utilizadas na classificação automática de cromossomos são o comprimento e posição do centrômero (ou índice centromérico), sendo que com estas duas características é possível separar os cromossomos em 7 grupos, também chamado grupo de Denver (Graham, 1987). Uma outra característica, talvez a mais explorada desde o advento da preparação de amostras de cromossomos com bandas, é o padrão de banda do cromossomo ao longo do eixo longitudinal. Estas três características tem sido a base da grande maioria dos estudos desenvolvidos nas duas últimas décadas (Desinov, 1994; Errington, 1993; Graham, 1987; Groen et al, 1989; Keret, 1991; Lundsteen, 1986 e 1981; van Vliet et al, 1990).

O problema da aplicação nesta fase, consiste em, fornecidas as três características de um cromossoma, identificar a qual dos grupos de Denver, ele pertence.

5.3.2 Comentários

Nos testes com esta base de dados, o objetivo inicial foi o de generalizar o método em relação à aplicação em problemas com diferentes

números de classes e características. Cumprida esta etapa de generalização do método, o objetivo se converteu em melhorar os resultados iniciais obtidos através da escolha de funções de pertinência adequadas e formas de otimizar as funções aprendizado e similaridade.

Foi desenvolvida uma aplicação com sete neurônios, um para cada um dos grupos de Denver, cada um com três características.

5.3.3 Resultados e comparações

Destacam-se a seguir os dois resultados mais significativos obtidos.

O primeiro obteve taxa de erro de 4,77% na base de teste com a seguinte configuração:

- intervalo de discretização com 100 elementos;
- parâmetro de combinações de características $H = 1$;
- número de combinações de características utilizado = 27;
- raio da função difusa = 2;
- função de pertinência triangular;
- número de épocas = 3.

O segundo, com a configuração abaixo, obteve 4,88% de taxa de erro na base de teste:

- intervalo de discretização com 100 elementos;
- parâmetro de combinações de características $H = 2$;
- número de combinações de características utilizado = 27;
- raio da função difusa = 1;
- função de pertinência triangular;
- número de épocas = 1.

Nesta aplicação, FAN obteve resultado inferior à RBF por Todesco [TODESCO1995], entretanto superou o resultado obtido por Henrington com MLP Backpropagation, apresentado em Todesco [TODESCO1995], conforme a tabela 5.1:

MODELO	TESTE (%)
Free Associative Neurons	4,77
Multi-Layer Perceptron Backpropagation (Henrington)	5,40
Radial Basis Functions (Todesco)	4,04

Tabela 5.1 – Comparativo entre taxas de erros (TESTE)

5.4 Terceira fase - Saúde financeira de pequenas empresas

5.4.1 Definição do problema

O problema consiste em determinar a situação de uma empresa dentre sete categorias ou classes possíveis:

- Lucratividade;
- liquidez a curto prazo;
- dívidas;
- sujeita a outras análises;
- lucratividade e liquidez a curto prazo;
- lucratividade e dívidas;
- liquidez a curto prazo e dívidas.

As características utilizadas na classificação são taxas financeiras de empresas. Segundo Pacheco [PACHECO1996], a extração das taxas

significativas, para serem utilizadas na determinação do diagnóstico, é mais difícil que o problema específico de classificação. Na abordagem nesta fase, aproveitou-se a elaboração usada pelo referido autor, que definiu oito características divididas da seguinte forma:

Taxas financeiras relacionadas aos problemas de lucratividade:

- cash flow to sales;
- net income to sales;
- working capital turnover.

Taxas financeiras aos problemas de liquidez a curto prazo

- cash conversion cycle;
- t-working investment TWI;
- TWI trend.

Taxas financeiras relacionadas aos problemas de débito

- earning before interest and taxes to investments;
- debt to equity.

Pacheco[PACHECO1996] em seu trabalho optou por gerar exemplos randomicamente, justificando-o. A distribuição utilizada pelo autor é a mesma na qual FAN foi aplicado, sendo que 200 empresas constituem a base de treinamento e 105 de testes.

5.4.2 Comentários

O objetivo nesta fase foi avaliar o desempenho do método, com o menor número de alterações possível em relação à aplicação da fase anterior.

Criou-se uma implementação com sete unidades FAN, cada uma representando uma das classes possíveis de situações de saúde financeira. Cada classe possui oito características de entrada, representando as taxas financeiras.

Vários testes foram realizados alterando-se os parâmetros do algoritmo e o número de épocas de treinamento.

5.4.3 Resultados e comparações

Destacam-se a seguir dois dos resultados mais significativos.

O primeiro obteve taxa de erro de 27,62% na base de teste com a seguinte configuração:

- intervalo de discretização com 17 elementos;
- parâmetro de combinações de características $H = 1$;
- número de combinações de características utilizado = 27;
- raio da função difusa = 1;
- função de pertinência triangular;
- número de épocas = 15.

O segundo, com a configuração abaixo, obteve 28,57% de taxa de erro na base de teste:

- intervalo de discretização com 20 elementos;
- parâmetro de combinações de características $H = 2$;
- número de combinações de características utilizado = 27;
- raio da função difusa = 1;
- função de pertinência triangular;
- número de épocas = 2.

(Obs. O resultado obtido na primeira configuração, é repetido na Segunda. Mesmo que se aumente o número de épocas de treinamento, este resultado se mantém.)

Em ambas configurações, o treinamento exaustivo (grande número de épocas) levou a taxa de erro a anular-se, em relação ao conjunto de treinamento, não alterando a taxa de erro obtida na base de teste.

Nesta aplicação, FAN obteve resultados mais satisfatórios que aqueles obtidos através da utilização das redes neurais MLP com Backpropagation e RBF como mostra a tabela 5.2 com as respectivas taxas de erro.

MODELO	TREINAMENTO (%)	TESTE (%)
Free Associative Neurons	0	28
Multi-Layer Perceptron	0	39
Radial Basis Function	11	42

Tabela 5.2 – Comparativo entre taxas de erros (TREINAMENTO e TESTE)

5.5 Experimentos

5.5.1 Experimentos relacionados à confusão entre as classes

Objetivo de realizarem-se os testes com as matrizes de confusão entre os dados reais e os apontados por FAN (Quadros 5.3 a 5.14) é verificar a importância do número de dados de treinamento e sua distribuição, para o bom

desempenho do método. Foram realizados 6 experimentos; cada um deles, gerou 2 matrizes:

- matriz de confusão dos resultados dos testes no próprio conjunto de treinamento escolhido;
- matriz de confusão no teste do treinamento sobre o conjunto de teste completo usado na aplicação da fase 2.

Os conjuntos de treinamento definidos para os testes foram gerados (exceto o conjunto completo), extraíndo-se um número pré-definido de amostras para cada classe do conjunto de treinamento (o mesmo da aplicação 2) de forma aleatória. Os conjuntos de treinamento escolhidos foram:

- o próprio conjunto de treinamento com 4061 padrões (aplicação 2) (Quadros 5.3 e 5.4);
- um conjunto com 2100 padrões distribuídos eqüitativamente (i.e. 300 para cada classe) (Quadros 5.5 e 5.6);
- um conjunto com 1400 padrões com 200 para cada classe (Quadros 5.7 e 5.8);
- um conjunto com 700 padrões com 100 padrões para cada classe (Quadros 5.9 e 5.10);
- um conjunto com 1600 padrões sendo distribuídos 200 para cada uma das 5 primeiras classes e 300 para as duas últimas (Quadros 5.11 e 5.12);
- um conjunto com 2100 padrões distribuídos entre as classes nas mesmas proporções do conjunto completo do treinamento (Quadros 5.13 e 5.14).

A seguir são apresentados os resultados de cada um dos testes.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	505	98,06	2	0,38	8	1,55	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	2	0,58	238	97,69	6	1,73	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	9	0,66	20	1,47	1312	96,47	16	1,18	3	0,22	0	0,00	0	0,00
Classe 4	0	0,00	0	0,00	8	1,50	516	96,45	11	2,06	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	1	0,19	4	0,74	529	98,14	1	0,19	4	0,74
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	20	5,56	312	86,67	28	7,78
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	4	1,00	41	10,10	361	88,92
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
4.061					3.873					4,63				

Quadro 5.1 – Teste realizado com treinamento completo sobre o próprio conjunto de treinamento

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	504	98,63	1	0,20	6	1,17	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	8	2,31	329	95,09	9	2,60	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	14	1,03	15	1,10	1316	96,91	11	0,81	1	0,07	0	0,00	1	0,07
Classe 4	0	0,00	0	0,00	9	1,70	507	95,66	14	2,64	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	4	0,75	11	2,06	509	95,14	7	1,31	4	0,75
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	35	9,72	303	84,17	22	6,11
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	5	1,23	33	8,15	367	90,62
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
4.045					3.835					5,19				

Quadro 5.2 – Treinamento completo aplicado ao conjunto de teste

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	294	98,00	3	1,00	3	1,00	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	2	0,67	296	98,67	2	0,67	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	4	1,33	6	2,00	286	95,33	4	1,33	0	0,00	0	0,00	0	0,00
Classe 4	0	0,00	0	0,00	3	1,00	296	98,67	1	0,33	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	0	0,00	4	1,33	291	97,00	2	0,67	3	1,00
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	20	6,67	263	87,67	17	5,67
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	3	1,00	35	11,67	262	87,33
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
2.100					1.988					5,33				

Quadro 5.3 – Teste realizado com treinamento com 2100 padrões, 300 por classe, aplicado sobre o próprio conjunto de treinamento

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	496	97,06	9	1,76	6	1,17	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	8	2,31	333	96,24	5	1,45	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	16	1,18	55	4,05	1265	93,15	18	1,33	3	0,22	0	0,00	1	0,07
Classe 4	5	0,94	0	0,00	4	0,75	510	96,23	11	2,08	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	3	0,56	21	3,93	498	93,08	9	1,68	4	0,75
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	31	8,61	304	84,44	25	6,94
Classe 7	0	0,00	0	0,00	0	0,00	1	0,25	5	1,23	38	9,38	361	89,14
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
4.045					3.767					6,87				

Quadro 5.4 – Treinamento realizado com 300 padrões para cada classe, testado no conjunto completo de testes.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	198	99,00	1	0,50	1	0,50	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	2	1,00	194	97,00	4	2,00	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	0	0,00	2	1,00	196	98,00	2	1,00	0	0,00	0	0,00	0	0,00
Classe 4	0	0,00	0	0,00	2	1,00	196	98,00	2	1,00	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	0	0,00	0	0,00	198	99,00	2	1,00	0	0,00
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	9	4,50	177	88,50	14	7,00
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	1	0,50	20	10,00	179	89,50
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
1.400					1.338					4,42				

Quadro 5.5 – Teste realizado com 200 padrões para cada classe, sobre o próprio conjunto de treinamento.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	500	97,85	8	1,57	2	0,39	1	0,20	0	0,00	0	0,00	0	0,00
Classe 2	9	2,60	328	94,80	9	2,60	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	18	1,33	21	1,55	1299	95,66	18	1,33	1	0,07	0	0,00	1	0,07
Classe 4	1	0,19	0	0,00	6	1,13	503	94,91	18	3,40	0	0,00	2	0,38
Classe 5	0	0,00	0	0,00	4	0,75	11	2,06	506	95,58	10	1,87	4	0,75
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	28	7,78	307	85,28	25	6,94
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	4	1,00	38	9,38	363	89,62
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
4.045					3.806					5,91				

Quadro 5.6 – Teste realizado com 200 padrões para cada classe, sobre o conjunto de teste.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	99	99,00	0	0,00	1	1,00	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	1	1,00	98	98,00	1	1,00	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	0	0,00	0	0,00	100	100,0	0	0,00	0	0,00	0	0,00	0	0,00
Classe 4	0	0,00	0	0,00	1	1,00	95	95,00	4	4,00	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	0	0,00	0	0,00	96	96,00	1	1,00	3	3,00
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	2	2,00	91	91,00	7	7,00
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	0	0,00	12	12,00	88	88,00
Número de Padrões de Treinamento 700					Número Total de Acertos 667					Taxa de Erro(%) 4,71				

Quadro 5.7 – Teste realizado com 100 padrões para cada classe, sobre o próprio conjunto de treinamento.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	497	97,26	8	1,57	5	1,00	1	0,20	0	0,00	0	0,00	0	0,00
Classe 2	10	2,89	324	93,64	12	3,47	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	28	2,06	51	3,76	1262	92,93	12	0,88	4	0,29	0	0,00	1	0,07
Classe 4	0	0,00	0	0,00	16	3,02	499	94,15	13	2,45	1	0,19	1	0,19
Classe 5	0	0,00	0	0,00	0	0,00	20	3,74	496	92,71	16	2,99	3	0,56
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	30	8,33	317	88,06	13	3,61
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	4	1,00	61	15,06	34	83,95
Número de Padrões de Treinamento 4.045					Número Total de Acertos 3.735					Taxa de Erro(%) 7,66				

Quadro 5.8 – Teste realizado com 100 padrões para cada classe, sobre o conjunto de teste.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	197	98,50	1	0,50	2	1,00	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	1	0,50	193	96,50	6	3,00	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	1	0,50	3	1,50	194	97,00	2	1,00	0	0,00	0	0,00	0	0,00
Classe 4	0	0,00	0	0,00	2	1,00	198	99,00	0	0,00	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	0	0,00	2	1,00	193	96,50	4	2,00	1	0,50
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	17	5,67	256	85,33	27	9,00
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	3	1,00	30	1,00	267	89,00
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
1.600					1.498					6,38				

Quadro 5.9 – Teste realizado com 200 padrões nas cinco primeiras classes e 300 nas duas últimas, sobre o próprio conjunto de treinamento.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	497	97,26	8	1,57	6	1,17	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	4	1,16	324	93,64	18	5,20	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	19	1,40	24	1,77	1280	94,26	32	2,36	2	0,14	0	0,00	1	0,07
Classe 4	0	0,00	0	0,00	5	0,94	515	97,17	8	1,51	1	0,19	1	0,19
Classe 5	0	0,00	0	0,00	4	0,75	19	3,55	493	92,15	15	2,80	4	0,75
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	29	8,06	308	85,56	23	6,39
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	4	0,99	37	9,14	364	89,88
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
4.045					3.781					6,53				

Quadro 5.10 – Teste realizado com 200 padrões em cada uma das cinco primeiras classes e 300 para as duas últimas, sobre o conjunto de teste.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	260	97,74	1	0,38	4	1,50	1	0,38	0	0,00	0	0,00	0	0,00
Classe 2	1	0,56	174	92,21	4	2,23	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	3	0,43	16	2,28	673	95,73	9	1,28	2	0,28	0	0,00	0	0,00
Classe 4	0	0,00	0	0,00	3	1,08	270	97,47	4	1,44	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	0	0,00	0	0,00	276	98,92	0	0,00	3	1,08
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	17	9,14	158	84,95	11	5,91
Classe 7	0	0,00	0	0,00	0	0,00	0	0,00	1	0,47	18	8,57	191	90,95
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
2.100					2.002					4,67				

Quadro 5.11 – Teste realizado com a distribuição de padrões proporcional à distribuição do conjunto de treinamento completo, sobre o próprio conjunto de treinamento.

FAN	Classe 1		Classe 2		Classe 3		Classe 4		Classe 5		Classe 6		Classe 7	
REAL	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%	Qtd.	%
Classe 1	498	97,46	2	0,39	11	2,15	0	0,00	0	0,00	0	0,00	0	0,00
Classe 2	8	2,31	329	95,09	9	2,60	0	0,00	0	0,00	0	0,00	0	0,00
Classe 3	13	0,96	33	2,43	1294	95,29	15	1,10	2	0,15	0	0,00	1	0,07
Classe 4	0	0,00	0	0,00	10	1,89	506	95,47	14	2,64	0	0,00	0	0,00
Classe 5	0	0,00	0	0,00	2	0,37	11	2,06	510	95,33	8	1,50	4	0,75
Classe 6	0	0,00	0	0,00	0	0,00	0	0,00	31	8,61	310	86,11	19	5,28
Classe 7	0	0,00	0	0,00	0	0,00	1	0,25	6	1,48	38	9,38	360	88,89
Número de Padrões de Treinamento					Número Total de Acertos					Taxa de Erro(%)				
4.045					3.807					5,88				

Quadro 5.12 – Teste realizado com a distribuição de padrões proporcional à distribuição do conjunto de treinamento completo, sobre o conjunto de testes.

Os testes realizados nestes experimentos não são conclusivos mas trazem informações que permitem compreender melhor o desempenho do método. O comportamento do aprendizado em relação às características do problema, trazem informações também a respeito da estrutura de cada classe.

Em todos os testes notou-se que o maior problema no reconhecimento, tanto no que se refere ao conjunto geral de teste quanto no próprio conjunto de treinamento de cada experimento, ocorre nas classes 6 e 7, principalmente na 6. Os resultados, nestas duas classes, praticamente não variam nos diversos experimentos. Nas demais classes, as taxas de erro, na maioria dos testes se mostraram inferiores à taxa de erro geral.

O teste realizado com 100 padrões para cada classe (Quadros 5.7 e 5.8), gerou a maior taxa de erro no teste no conjunto completo de teste, o que indica a diminuição da capacidade de generalização nos aprendizados com conjuntos pequenos de treinamento. Porém, o aprendizado do próprio conjunto de treinamento é comparável os resultados obtidos nos conjuntos maiores.

A distribuição que gerou os melhores resultados no reconhecimento do próprio conjunto de treinamento foi a que usou 200 padrões distribuídos igualmente entre as classes (Quadros 5.5 e 5.6). Com exceção do teste com o conjunto completo de treinamento, este experimento obteve a menor taxa de erro também em relação ao conjunto de teste. Estes resultados determinaram a escolha deste conjunto para a realização dos testes com o número de épocas.

O teste realizado com o treinamento com 200 padrões para as cinco primeiras classe e 300 para as duas últimas, teve como objetivo reforçar as classe mais problemáticas, para verificar a possibilidade de reverter os resultados negativos. Como pode ser observado no Quadro 5.10, este experimento não

logrou êxito, uma vez que a taxa de erro no conjunto de teste aumentou em relação a distribuição com o mesmo número de padrões distribuídos de forma igual para cada classe.

No mesmo sentido, foi realizado o experimento com a distribuição dos padrões guardando as proporções do conjunto completo de treinamento. A taxa de erro neste caso (Quadro 5.12), foi maior que no experimento com o mesmo número de padrões distribuídos de forma igualitária entre as classes (Quadro 5.3). Isto sugere que para o método proposto, a proporção na distribuição do número de padrões entre as classes não é significativa.

A menor taxa de erro dos experimentos foi a obtida com o conjunto completo de treinamento, o que implica que a capacidade de generalização aumenta com o aumento do número de protótipos de treinamento (diversidade).

De forma geral, pode-se dizer que todos os experimentos apresentaram resultados satisfatórios e semelhantes, considerando-se as circunstâncias em que foram realizados. O principal fator apontado pelos testes no que se refere a diminuição da taxa de erro no conjunto de testes (aumento na capacidade de generalização) é o número de padrões no conjunto de treinamento, enquanto que na capacidade de reconhecimento do próprio conjunto de treinamento este critério não é relevante.

5.5.2 Experimentos relacionados ao número de épocas

O objetivo deste experimento é estudar o comportamento da taxa de erro resultante dos testes dos treinamentos no conjunto de teste e no próprio conjunto de treinamento. No primeiro caso, avalia-se a capacidade de generalização. No segundo, o aprendizado de um objetivo específico (reconhecer

o próprio conjunto de treinamento). O conjunto de treinamento utilizado para este propósito é formado por 1400 padrões distribuídos igualmente entre as classes (o mesmo dos Quadros 5.5 e 5.6). O gráfico da Figura 5.1, ilustra este comportamento.

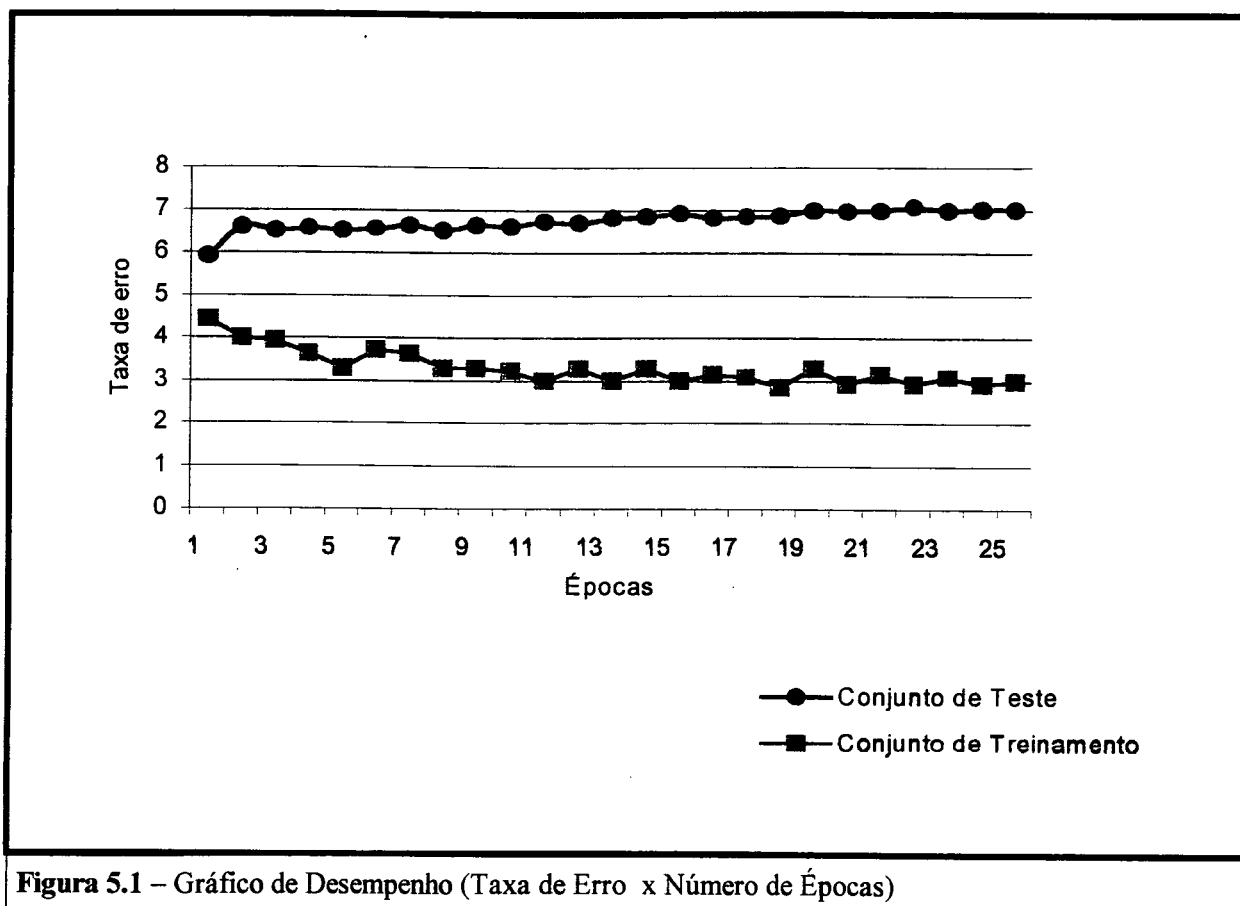


Figura 5.1 – Gráfico de Desempenho (Taxa de Erro x Número de Épocas)

No gráfico pode ser observado que o treinamento repetido provoca uma estabilização, tanto na taxa de erro obtida no teste sobre o conjunto de teste quanto no teste realizado sobre o próprio conjunto de treinamento. Isto quer dizer que a taxa de erro no conjunto de teste, provocada pelo treinamento exaustivo, não aumenta a partir de um determinado ponto. Parece haver uma métrica de similaridade entre o conjunto de treinamento e o de testes. Esta hipótese é uma possibilidade interessante para estudos futuros, não fazendo parte do escopo deste trabalho.

6 Conclusão

Um Neurônio Livre Associativo é uma unidade composta por um conjunto de parâmetros que representa um modelo. Possui um algoritmo de treinamento, que aprende as características deste modelo e outro, que ao receber um conjunto qualquer de características, atribui o seu grau de pertinência em relação ao conjunto de características que definem o modelo.

Em um sistema de reconhecimento de padrões (RP), cada classe pode ser associada a um modelo ou protótipo, que pode ser representado por um Neurônio Livre Associativo ou unidade FAN. A unidade aprende as características dos padrões associados à classe, aplicando um algoritmo FAN em um conjunto de padrões de treinamento. A associação de um padrão a uma determinada classe é feita através da escolha do neurônio (i.e. modelo/unidade FAN), à qual o padrão possui maior grau de pertinência em relação ao seu conjunto de características.

O método apresentado neste trabalho mostrou-se perfeitamente adaptável a diferentes categorias de problemas de RP. Os testes produziram bons resultados nas aplicações realizadas, entretanto por se tratar de uma abordagem nova, há muito ainda para ser estudado e novos caminhos podem ser vislumbrados.

A característica de FAN possuir unidades de representação independentes para as classes de um problema de reconhecimento de padrões, torna-o um método capaz de utilizar diversos conjuntos de treinamento diferentes conjuntamente (um FAN para cada treinamento), e usar técnicas difusas para

definir com maior precisão, a pertinência de um padrão de teste ao conjunto de características que definem uma classe (i. e. melhorar a capacidade de classificação). Por exemplo, pode-se supor três conjuntos diferentes de treinamento: T_1 , T_2 e T_3 , com padrões de uma classe específica C . Para um padrão x podemos obter:

$$\begin{aligned} FAN_{T1}(x) &= \mu_{1x}(C) \\ FAN_{T2}(x) &= \mu_{2x}(C) \\ FAN_{T3}(x) &= \mu_{3x}(C) \text{ e} \\ \mu_x(C) &= \max(\mu_{1x}(C), \mu_{2x}(C), \mu_{3x}(C)) \end{aligned}$$

onde $\mu_{ix}(C)$ representa o grau de pertinência do padrão x , ao conjunto de características que definem C .

O exemplo acima pode ser aplicado na busca da solução para o problema da saturação no processo de treinamento que ocorre ao se utilizar redes neurais. Depois de realizadas muitas épocas de treinamento, as redes acabam *viciando* no conjunto de treinamento, perdendo a capacidade de generalização necessária para o bom desempenho nas aplicações reais. A proposta para os trabalhos futuros nesta linha, é a de que os conjuntos de treinamentos sejam divididos em sub-conjuntos e que vários treinamentos sejam realizados independentemente, gerando várias unidades FAN para cada classe (quantas se julgar necessário), que combinadas entre si, podem melhorar o desempenho na aplicação.

Outra característica importante do método tratado, é a sua capacidade para generalizar a informação com poucos dados de treinamento. A queda na performance constatada nos experimentos realizados em treinamentos com um número pequeno de padrões, apesar de ser significativa em relação aos melhores resultados obtidos, não foi exagerada. Pela característica do método (unidades e parâmetros independentes), um sistema ao utilizar FAN possui boa

capacidade de reação às mudanças dos dados no treinamento. Além disto em qualquer tempo durante o treinamento, o algoritmo FAN pode fazer uma projeção ou sugestão da classe à qual o padrão apresentado pertence com maior possibilidade. A soma destas condições em um método de RP, deve permitir obter êxito na solução de problemas com treinamento e teste dinâmicos que também pode ser uma área fértil para o desenvolvimento de trabalhos futuros.

Ainda há outras recomendações para trabalhos que poderiam ser desenvolvidos, tais como:

- buscar a evolução na estrutura genérica de FAN;
- desenvolver bases de conhecimento baseados em FAN;
- utilizar FAN para controle;
- aplicar FAN em outros problemas práticos conforme surgir demanda.

Os principais problemas apresentados pelo método, se referem às limitações dos recursos computacionais. A rapidez no treinamento de um sistema FAN, mostra-se inferior às técnicas que utilizam algoritmos de redes neuronais.

Finalmente, podemos concluir que FAN é uma técnica que reúne a capacidade de generalização e adaptação dos métodos de RP difuso, com a vantagem de produzir automaticamente o seu aprendizado, de forma semelhante aos processos que ocorrem nas redes neuronais. Sendo assim, pode-se tratá-lo como uma nova abordagem a ser aplicada e estudada.

Referências Bibliográficas

- BASTOS, Rogério C. *Avaliação de Desempenhos Educacionais: uma Abordagem usando Conjuntos Difusos*. Tese de Doutorado. Universidade Federal de Santa Catarina. Departamento de Engenharia de Produção, 1994.
- BISCHOP, Christopher M. *Neural Networks for Pattern Recognition*. Oxford University press, 1995.
- CICHOCKI A.; UNBENHABEN, R. *Neural Networks for Optimization and Signal Processing*. John Wiley & Sons inc, 1993.
- DANDOLINI, Gertrudes A. *Um procedimento para avaliação da saúde financeira de pequenas empresas: estudo de um caso usando redes neuronais artificiais*. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Departamento de Engenharia de Produção, 1996.
- FISHER, R.A. *The Use of Multiple Measurements in Axonomic Problems*. 1936. [doc WWW Data: 15/10/1997].
<http://www.comcat.com/~hutch/DASL/dataex.htm>
- FU, L. M. *Rule Generation from Neural Networks*. IEEE Transactions on Systems. Man and Cybernetics. Vol. 28. No. 8. pp. 1114 - 1124, 1994.
- FUKUSHIMA, K.; MIYAKE, S.; ITO, T. *Neocogntron: A Neural Network Model for a Mechanism of Visual Pattern Recognition*. IEEE Transaction on Systems. Man, and Cybertics SMC-13. pp. 826-834, 1983.
- GOLDBERG, David E. *Genetic Algorithmis in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
- GOONATILAKE, S.; KHEBBAL, S., *Intelligent Hybrid Systems*. Proceedings of First International Conf on Intelligent Systems. Singapore. 1992.
- GOONATILAKE, S.; Khebbal, S. *Intelligent Hybrid Systems: Issues, Classifications, and Future Directions*. in [Goonatilake & Khebbal, 1995a].
- HAYKIN, Simon. *Neural Networks: A comprehensive Foundation*. Macmillan College Publish Company. New York, 1994.

- JOHNSON, Richard A.; WICHERN, Dean W. *Applied Multivariate Statistical Analysis*. Prentice Hall inc., 1982.
- KANDEL, A. e LANGHOLZ, G., *Hybrid Architectures for Intelligent Systems*, Ed. CRC Press. Inc., 1992.
- KLIR, G. J. & YUAN, B. *Fuzzy sets and fuzzy logic - theory and applications*. Prentice Hall, 1995.
- KOSKO, B. *Neural Networks and Fuzzy Systems*. Prentice Hall, Englewood Cliffs, NJ, 1992.
- LAPOLLI, Edis M. *Processamento de Imagens Digitais: Uma Abordagem utilizando conjuntos Difusos*. Tese de Doutorado, Universidade Federal de Santa Catarina, Departamento de Engenharia de Produção, 1994.
- LEV, B., and SUNDER, S. *Methodological issues in the use of financial ratios*. Journal of Accounting and Economics 1/3,(1979) 187-210.
- LEV, B., *Financial Statement Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1974.
- MCNEILL, D.,FREIBERGER P. *Fuzzy Logic: The Revolutionary Technology that is Changing our World*.Touchstone, 1993.
- MARTINS, A. R.. *Modellinf Financial Statements Analysis for Indicating Correstive Actions to Financial Problems*. Tese de Doutorado, Universidade Federal de Santa Catarina, Departamento de Engenharia de Produção, 1996.
- MARTINS, A.; PACHECO, R.; WEBER-LEE, R.; BARCIA, R. M. *Integrating Expert Networks and CBR in a Hybrid Architecture to Analyze the Financial Health of a Firm*, Brazil 2nd International Congress of Industrial Engineering and 16th National Congress of Production Engineering - ENEGEP , October, 7 - 10, Piracicaba, São Paulo, Brasil, 1996.
- MEDSKER, Larry. *Hibrid Inteligent Sístems*. Boston: Kluwer Academic Publishers, 1995.
- PACHECO, R. C. S.. *A Hybrid Intelligent System for Diagnosing and Solving Financial Problems of Small Retail Firms*, Ph.D. Dissertation, Universidade Federal de Santa Catarina UFSC, Programa de Pós-graduação em Engenharia de Produção, Brasil, 1996.
- RAITZ, R. T.; SOUZA, J.A.; DANDOLINI, G. A.; PACHECO, R. C. S.; MARTINS, A., GAUTHIER, F & BARCIA, R.. Learning by Means of Free

- Associative Neurons. NAFIPS 97: *Annual Meeting Of The North American Fuzzy Information Processing Society*, September 21-24, 1997, Syracuse, New York.
- ROSS, Timoty J. *Fuzzy Logic with Engeneering Applications*. McGrow-Hill inc., 1995.
- SCHALKOFF, Robert J., *Pattern Recognition: Statistical, Structural and Neural Approches*, John Wiley & Sons, Inc., 1992.
- TODESCO, Jose L. *Reconhecimento de Padrões Usando Rede Neuronal Artificial com uma Função de Base Radial: uma Aplicação na Classificação de Cromossomas Humanos*. Tese de Doutorado, Universidade Federal de Santa Catarina, Departamento de Engenharia de Produção, 1995.
- WEBER, Rosina de O. *Sistema Especialista Difuso Para Análise De Crédito*. Dissertação de Mestrado, Universidade Federal de Santa Catarina UFSC, Programa de Pós-Graduação em Engenharia de Produção, Florianópolis, 1993.
- ZADEH, L.A., *Fuzzy sets*. Information and Control, Vol. 8, 1965, pp. 338-353.
- ZADEH, L.A., *Fuzzy algorithms*, Info. & Ctl., Vol. 12, 1968, pp. 94-102.
- ZADEH, L.A., *Making computers think like people*, IEEE Spectrum 8, pp. 26-32, 1984.